Humanoids toward Practical Applications - Robotic Material Handling

A Thesis

Submitted to the Faculty

of

Drexel University

by

Youngbum Jun

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy in Mechanical Engineering and Mechanics

April 2014

ABSTRACT

Humanoids toward Practical Applications - Robotic Material Handling

Youngbum Jun
Advisor: Paul Y. Oh, Ph.D.

Humanoids have been developed over the past few decades through the integration of many high-level technologies. These robots are designed to function in human-centered environments, like homes, offices, hospitals and factories. Previous generations of humanoids have been developed to showcase technological prowess and captivate the general public but the demand for *practical* uses like patient nursing in hospitals, industrial efficiency in factories and warehouses, and disaster restorations has spurred research towards more robust interactions with humanoids and human-centered environments. Such practical uses require humanoids' capabilities to *percept* surroundings, *manipulate* tools, and *navigate* through the real world environments. Furthermore, these humanoids must be *reliable*, *stable*, and *adaptable* to arbitrary environments. However, a lack of artificial intelligence and integration of loosely-coupled sub-fields in humanoids, namely legged walking, motion planning, force control and manipulation, hinder the immediate utilization of these robots in practical applications. This dissertation demonstrates a *tele-operated* humanoid platform which shares autonomy with human operators. The semi-autonomous humanoid receives high-level commands, such as decision making, from human operators but controls low-level functions, such as balance, locomotion and manipulation. The methodologies described in this dissertation are theoretically constructed and empirically tested and evaluated on a full-sized humanoid. Through the 2013 DARPA Robotics Challenge-hose installation and rough terrain walking challenges, humanoids' capabilities to

function within a human-centered environment have demonstrated.

Table of Contents

List of Tables

List of Figures

1. INTRODUCTION

Humanoids are anthropomorphic robots and thus should be able to move in *human-centered* environments and handle *human-operated* tools. With such movement and handling, humanoids could have been used in the Fukushima nuclear accident. These robots would be able to open and walk through doors, climb ladders and operate wall-breaking tools and climb ladders. These robots would turn valves, flip switches and connect hoses. With such locomotion and manipulation capabilities, much of the radiation could have been contained in the early stages of the reactor leak to prevent the catastrophic meltdown. Despite having some of the most advanced humanoids in the world near Fukushima, attempts to deploy them were not even considered; despite their human-form, they were ill-equipped to robustly perform practical human-functions.

Fukushima could be viewed as an indictment on the humanoid state-of-readiness. But rather than a condemnation, the state-of-art reflects what has been prioritized. Over the past two decades, full-sized humanoids were designed to showcase and incubate technologies. Towards this, humanoids walked, ran, danced and played musical instruments to the delight of public audiences, the media and policy-makers. How such priorities, effort and successes translate to practical and meaningful work has always been a matter of debate within the robotics community. As such, Fukushima can be viewed as a *wake-up* call to channel the field's successes and efforts towards enabling humanoids to do practical locomotion and manipulation tasks.

Identifying a starting point for what is deemed *practical* is both challenging and debatable. Perhaps a conceptual approach reveals gaps and hence gives research direction. Most full-sized humanoids today are the size of a 10-year old boy. Conceptually such boys have little trouble opening common doors and climbing most ladders. Such

boys also can walk over most terrain without losing balance and can recover from minor trips. 10-year old's can also turn moderately tight valves, flip most switches and screw threaded hoses. Such notions provide a level of balance (for locomotion) and dexterity (for manipulation) needed to begin a discussion on what constitutes practical work. Such discussion helps establish some thresholds and objectives for humanoid design requirements.

This dissertation defines *practical* work as a humanoid that locomotes in human-centered environments and handles human-operated tools. For this dissertation, examples should go beyond smooth-floors and clutter-free sites and more demanding than picking up homomorphic objects. Humanoids like ASIMO have demonstrated the ability to locomote in carpeted wide-open office spaces and grasping cylindrical and rectangular objects like cups and small box packages. The vertical advance in the field demands examining less defined terrain that may demand modes like crawling, climbing over obstacles, and squeezing through holes in walls. The advance also demands investigating more complex objects like hand tools, fire hoses, and valves. Meeting these demands could then lead to practical tasks like turning valves, attaching hoses and breaching walls. Such tasks would advance robotic disaster-response. But moreover, such capabilities could be re-focused towards advancing sectors like manufacturing, construction, and agriculture where humanoids locomote in factory floors, work sites, and crop fields to operate pumps, demolish walls, and handle pesticides. Such sectors, like disaster-response involve tasks that are often dirty and dangerous but are important. A *practical* humanoid would thus have broad impacts removing people from hazards and harm.

The dissertation's *general objective* is to realize a humanoid for practical work. Arguably such an objective represents a "holy grail" for robotics – to mimic what a 10-year old boy can do would disrupt technologies and be transformative. Such

mimicking would demand advances in both the physical and cognitive capabilities of humanoids. Discussions on the state-of-the-art in Chapter. 2 will show that realizing practical work is at a "tipping point". To repeat the above, priorities in the past led to realizing public showcasing (dancing, running, and instrument playing). Consequently, research focused on balancing statically and quasi-statically like standing in place or while mid-air when running. Focus also centered in scripted motions in structured environments like executing trajectories to play musical instruments or grasp simple-shaped objects. Such efforts have yield marvelous systems integration but the priorities perhaps obscure more fundamental gaps that are preventing significant leaps forward.

A *critical gap* in the state-of-the-art is the lack of methods that account for un-modelled phenomena like compliance and both determinate and undeterminate joints. Such phenomena serve important roles in the way people interact with their environment and suggest they should be accounted for in humanoid whole-body control. Thus the dissertation's *specific objective* is **FILL IN THE GAP**.

The dissertation's intellectual merit stems from its integration of loosely coupled sub-fields in humanoids, namely legged walking, motion planning, force control and manipulation. This integration is centered on real-time modification of motion and control according to the changes of dynamics, kinematics, and stability influenced by interaction with environments. Another merit comes from its design objectives. The integrated system accounts for real-world phnomena like wind force and cluttered grounds. This system shared with human intelligent allows a humanoid to perform a task in outdoor circumstances. These works are fully verified and validated throuth both simulation and exeriments with a full-sized humanoid.

2. LITERATURE REVIEW

Humanoids have been developed over the last few decades with an integration of many high level technologies. Some examples are Honda's ASIMO [32][76][95], Japanese Humanoid Research Group's HRP-2 [51], Korea Advanced Institute of Science and Technology's Hubo [58], Korea Institute of Science and Technology's MAHRU [67], Waseda University's WABIAN [85], and Technical University of Munich's Johnnie [72].[1]. These robots have demonstrated preprogrammed performances like playing musical instruments, human-like walking, and choreographed dancing. Such showcased technologies have successfully led to greater public interest and concern about humanoids. Beyond that, in fact, these robots have the potential to assist, interact, and work in human-centered environments like homes, offices, factories, outdoor fields, etc. These robots have the potential to improve industrial warehouse efficiency and take care of household affairs. Even more, these robots can be deployed to disaster areas like Fukushima affected by the March 2011 Tsunami and New Orleans impacted by Hurricane Katrina to assist with restoration works and humanitarian aid. While the long term goal is to make these robots fully autonomous for such practical uses, it remains very challenging to implement this. *The practical uses of humanoids require an integration of* **perceptibility, maneuverability, and manipulability** *which must be* **reliable, stable, and adaptable** *to arbitrary environments with or without a priori knowledge.*

---

[1]Many miniature humanoids have been developed and commercially available. These robots have been actively used as a research tool; Hitec's Robonova, Kondo's KHR, Fujitsu's Hoap-2, Robotis' Darwin-Op, Virginia Tech's Mini-Hubo, Drexel University's ATLAS, Sony's QRIO, Aldebran's Nao, etc.

2.1   Current State-of-Arts

Perceptibility is a key ability for humanoids to autonomously maneuver and handle the tools and objects people use. Geometric information is necessary to plan kinematic motion and tactile sensing allows humanoids to dynamically interact with people and environments. Such perception perhaps does not only mean visual recognition, tactile sensing, and sound detection but also includes the understanding of robot's status and localization of people, buildings, cars, etc. Current robotic systems are capable of sensing at a very high level with increasingly complex environments. Unmanned Aerial Vehicles (UAV) are autonomously and remotely piloted to fly at near-earth environments [33][31]. Unmanned Ground Vehicles (UGV) drive in urban areas [109]. Bio-inspired robots and mobile manipulators traverse mountains and remove heavy debris [92]. These systems typically equip Global Positioning Systems (GPS), Inertial Measurement Units (IMU), High-Definition (HD) video cameras, and laser scanners to perceive and localize within environments. Tactile sensors measuring force and torque allow these robots to manage dynamic interactions with surroundings. Current humanoids do not fully utilize these sensors and perception techniques because of limited physical capabilities in joint torque, battery power, shape of the structure, etc. But, those may be eventually integrated as the advance of technologies.

With these levels of perceptibility, autonomous maneuvering can be realized. Humanoids are an anthropomorphic mobile manipulator. Biped locomotion is not always required but appropriate to roam freely in very dense surroundings like homes, offices, and factories. During locomoting, the arms can be used for holding items and handling objects. However, the implementation of such biped locomotion is a chellenge because the motion is very unstable. Planning stable biped gaits for locomotion on structured or unstructured terrains are fully constrained in terms of stability. Vuko-bratovic et. al. [111][110] introduced the concept of Zero-Moment Point (ZMP),

which is a point on the ground where the sum of the moments about sagittal and transverse axes are zero. The idea simply says that a humanoid is stable if the ZMP lies within the support polygon. Otherwise, the robot tips over. With the ZMP, the previous works for gait planning can be roughly classified into three groups according to the methods to derive the stable pelvis motion corresponding to the foot motion.

One group plans the locomotion using *motion primitives.* They define a set of the necessary pelvis and feet positions in locomotion. Each position is a way point that the robot follows. The timing is assigned to each motion primitive that keeps the ZMP within the support polygon. Then, such motion primitives are interpolated for a continuous gait pattern. With this approach, Zheng et. al. [118] demonstrated a half-body biped system walking on a slope. Ono et. al. [86] designed a gait pattern for walking on an unspecified incline. Huang et. al [35] applied the third-order spline interpolation to foot motion primitives corresponding to the reference ZMPs, and experimentally tested such motion on a slope [37]. Li et. al. [70] and Fu et. al. [20] realized a humanoid walking on stairs. Recently, such motion primitives are often taken directly from human motions using Motion Capture (MoCap) systems. Kim et. al. [62] captured a human stair climbing and mapped such captured motion primitives into a humanoid to walk on stairs. These studies successfully generated the stable biped gaits on even and uneven terrains. However, this approach requires a large motion primitive look-up table to realize many different gait patterns. Such gait patterns are often limited to a particular terrain. On the other hand, the other two groups derive the biped gaits from *differential equations.* They mathematically model the dynamics of a humanoid. A relationship between the motion of the Center of Mass (CoM) and ZMP is derived from the model. Such relationship is a set of differential equations referred to the ZMP equation. One group solves the ZMP equation analytically while another group solves it numerically. A Three-Dimensional

Linearized Inverted Pendulum Model (3D-LIPM) was proposed by Kajita et. al. [49], which simplifies the ZMP equation to a second-order differential equation. In the group of *analytic approach*, Noh et. al. [83] assumed the simplified ZMP equation as a second-order non-homogeneous differential problem with boundary conditions. The Finite Difference Method was used to analytically solve this problem. The resulted gait trajectories were tested on a half-body biped robot, IWR. Choi et. al. [14] directly applied Laplace Transform to solve this ZMP equation. From the exact solution in the time-domain, a periodic function for forward locomotion was obtained. Kurt et. al. [66] and Erbatur et. al. [18] applied Fourier methods to solve the ZMP equation in order to generate the forward walking gait. Park et. al. [89] represented biped motion as a periodic function using sinusoidal functions. The amplitude and frequency of such functions were analytically derived from the ZMP equation. The resulted biped walking patterns were tested on a full-sized humanoid, Hubo. With this periodic function approach, Kim et. al. [59] generated a gait for stair climbing, Kim et. al. [61] realized a humanoid walking with an energy-efficient stretched gait, Kim et. al. [60] demonstrated a humanoid walking on uneven terrain and inclined floor, and Cho et. al. [12] implemented a humanoid running on an even surface. Kim et. al. [57] solved the ZMP equation in the discrete-time domain. The gait pattern was produced using convolution sum method which takes the future and past ZMP reference values. Park et. al. [88] employed this method to simulate a miniature humanoid, Hoap-2, walking on uneven terrains in Webot simulation environment. With *numerical ways*, Kajita et. al. [47] regarded the biped gait planning as a feedback control problem and proposed an optimal ZMP preview control. This control problem had the reference ZMP trajectory as an input, ZMP equation as a plant, and CoM position as an output. A Linear Quadratic Integrator (LQI) with previewable references proposed by Katayama et. al. [53] was applied to track the reference ZMP

trajectory. With this method, Park et. al. [90] applied the whole-body ZMP equation to this control in order to produce a walking gait more reliable than that made with a simplified ZMP equation. Huang et. al. [38] simulated a humanoid walking on a slope. Dallali et. al. [15] modeled a humanoids as Double Inverted Pendulum (DIP) to utilize the upper body in locomotion. Jun et. al. [43] realized a real-time gait planner and applied it to a miniature humanoid for obstacle avoidance.

The environments may be covered with water, snow, or brush and the ground may be obscured. The uncertainties from sensory measurements and rigidity of the mechanical structure often hinder planning locomotion from perfectly matching the ground topography. Because of the lack of compliance in humanoids, such kinematic errors generate a large reactionary force that directly yields moments about the foot. When the foot lands in locomotion, the impact forces and moments by the errors have to be absorbed for humanoids to maneuver in human-centered environments. Previous efforts have focused on implementation of both passive and active compliances. The passive compliance is a mechanical design. It generally consists of physical springs, dampers, and switches. Such components absorb the energy in landing and detect foot contact. Yamaguchi et. al. [113] introduced a shock absorbing foot for the biped vehicle, WL-16. Kim et. al [63] designed a four point contact spring-damper foot to minimize the impact force during contact. Li et. al. [69] designed a flexible compliance foot containing rubber brushes. Zhang et. al. [117] proposed an elastic energy storage foot and Hashimoto et. al. [30] implemented bio-inspired wearable foot design. Instead of foot designs, Takahashi et. al. [107] focused on joints and realized impact relaxation joints in legs. On the other hand, active compliance virtually realizes the spring-damper system. Such virtual compliance is typically designed based on modeling dynamics of landing and feedback control laws. Huang et. al. [36] implemented landing timing control. Early landing and late landing cases had

a different control loops. Lim et. al. [71] applied impedance control laws [34] to introduce compliance into the impact force at landing. Sugahara et. al. [106] realized a virtual spring-damper system. Hashimoto et. al. [29] introduced a compliance control based on a non-linear landing model and applied it to the biped walking. Hwang et. al. [39] designed a fuzzy controller to allow compliance in ankle roll and pitch when walking on an inclined surface.

Handling disturbances is probably one of the core factors that evaluate the reliability of humanoids working in the real world. Such disturbances internally include mechanical vibration, stiffness, and deformation and are caused by environmental sources like wind and collisions with surrounding objects. Losing balance and unintended motion resulting from such disturbances might damage the robot itself, property, and people. The previous works have focused on the design of controllers for humanoids to properly respond to such disturbances. For managing internal disturbances, ZMP-based controllers have been proposed [50][82][81][52][13][102]. The control objective of these controllers was mostly to constrain the ZMP within the support polygon. Based on the force and torque measurements acting on the foot, these controllers directly controlled the ankle joint orientation or CoM motion. The other studies have realized momentum controllers [48][21][68][11]. With inertial sensors like IMUs and accelerometers, these controllers maintained the angular momentum of the CoM as zero and tracked a desired linear momentum. This idea has been used to compensate for external impulsive disturbances [101][75][73][115]. These works planned a motion of specific body parts like the arms and pelvis to recover the posture against external forces. Pratt et. al. [91] proposed an idea to step out for balancing against a large external disturbance. Mao et. al. [74] utilized continuous stepping so that the linear momentum caused by the disturbance dissipates gradually. In addition, Kim et. al. [59] designed a damping controller to attenuate the mechanical vibration and

Nagarajan et. al. [78] introduced a safety falling directional control for humanoids.

Humanoids are high Degree-Of-Freedom mobile dexterous manipulators which have four end-effectors (two hands and two feet). Each end-effector is represented by its position (X, Y, and Z) and orientation (Roll, Pitch, and Yaw) in Cartesian space. But, humanoids typically have more than 30 DOF. When the goal configurations of all end-effectors are set, humanoids may have more than one set of joint configurations (more than one solution) that satisfies such goal positions and orientations because of the redundant DOFs. Such redundant joint motions often result in collisions with the robot itself or objects in the environment and can lead to instability in manipulation. Khatib et. al. [55] and Nakamura et. al. [79] proposed a null-space projection method for unconstrained manipulators. These methods prioritized the tasks in joint space and optimized the solution with the redundant joints through Inverse Kinematics (IK) and Inverse Dynamics (ID). As a result, the redundant joints were controlled. Siciliano et. al. [100] generalized this idea for highly-redundant robotic manipulators. Sentis et. al. [97] applied these approaches to humanoids. This work used the joint redundancy for stability control and avoiding a singularity in humanoid manipulation tasks. Kuffner et. al. [65] introduced a randomized path planning algorithm called Rapidly-exploring Random Trees (RRTs). This study found a joint configuration which is collision-free and stable. Berenson et. al. [7] generalized this method and demonstrated humanoids performing multiple tasks simultaneously in [6]. The same method was used for humanoid valve turning by Alnunni et. al. [1]. Here, another interesting approach to realize humanoid whole-body motion has been done through motion capture techniques presented in Jenkins et. al. [40], Nakaoka et. al. [80], Shon et. al. [99], Ott et. al. [87], and Yamane et. al. [114]. These works captured human motions and directly mapped them to humanoid kinematics.

Humanoids will likely perform tasks like parcel delivery, maintenance work, vehi-

cle operation, etc., which require interaction with physical objects. Such interactions yield reactionary forces that often result in instability and undesired behaviors. Even people manage their postures and strength to handle such interactions according to the situations. In humanoids, many approaches have been proposed to deal with such reactionary forces in interaction. Harada et. al. [25] introduced two ZMPs, Generalized ZMP (GZMP) and current ZMP, in manipulation. The GZMP is a virtual ZMP where the current ZMP is displaced to by interactional forces. With the calculation of the ZMP displacement, the CoM controller moves the body to compensate it. The ideas were applied to pushing a heavy object [24] and carrying it [26]. Stilman et. al. [103][104] planned a force trajectory through learning algorithm. Impedance-controlled arms and CoM position controllers allowed a humanoid to move a wheeled table with a 55kg load. Nozawa et. al [84] designed dual-arm force controller. Both arms complied to and actively produced force to push a wheelchair. The force difference in arms were used to determine gait transitions to track a given path. Cha et. al. [10] investigated the force acting on a humanoid in vertical direction. This study proposed the Motion Embedded CoM Jacobian method to actively compensate such force by the CoM motion. Arisumi et. al. [2][3] focused on dealing with an impulsive force. This work demonstrated a humanoid lifting an object that cannot be moved by continuous force. The idea was also used to open a door [4]. In addition, many previous works have been done to control the interactive force in human-humanoid collaboration [116][19][22][105], humanoid-humanoid collaboration [56][77], and tool handling [108].

2.2 Critical Gap

Again, humanoids have a form-factor to function in human-centered environments. Recently, demands for these robots working for industrial efficiency and dis-

aster restoration have been exponentially increasing. The research in humanoids also has been shifted from the preprogrammed performance to practical uses. Current state of arts reviewed here demonstrate humanoids' capabilities to locomote within structured or unstructured environments, manipulate tools and objects, and even collaborate with humans or other humanoids. *The* **critical gap** *toward humanoids working in such applications is that the performance of these robots is often limited to a particular environment like offices and laboratories. Real-world outdoor phenomena like wind, weather, sunlight, cluttered terrains, etc. has not been fully taken into account. Moreover, current state of arts are often focused on the performance of fully autonomous modes while the current mode of artificial intelligent development cannot catch up with such advances in performance.*

2.3 Filling the Gap

Under NSF PIRE project, the author has concentrated on dynamic motion planning and control of humanoid whole body manipulation and locomotion since 2008. Toward humanoids autonomous maneuvering, the author, Jun et. al. [43], implemented a humanoid for obstacle avoidance. In this work, a real-time biped walking planner was realized using the optimal ZMP preview control algorithm [47]. The gait pattern was planned in real time based on visual information from a camera. The observer-based ZMP controller stabilized the robot during walking. This work showed a humanoids' ability to autonomously locomote and follow a given path on even terrain. In 2011, the author, Jun et. al. [41], realized a humanoid walking on uneven terrain. In this study, the author investigated locomotion planned through analytic and numerical methods. Both methods were combined together in order to eliminate undesirable linear momentum causing instability in walking. Analytic vector Inverse Kinematics was also developed, which allows a humanoid faster control in balancing.

This study was fully verified and validated through a 3-tier infrastructure; a research tool which the author developed. For whole-body manipulation, the author, Jun et. al. [42], investigated the relationship between the posture and reactionary force in manipulation. In this work, a bio-mechanic study of human postures in pushing was done and such postures were realized on a humanoid. From mathematic analysis of postures in pushing, a posture that maximizes the physical capability of humanoids was introduced. The proposed posture controller was designed to achieve that posture.

Under the project for the 2013 DARPA Robotic Challenge, the author's experience, knowledge, and skills were dedicated to realize the performance of humanoids in human-centered environments. As a leader of the Obstacle Challenge [44][112][119], and Hose Installation Task [45][46][16], the author fully integrated maneuverability, manipulability, and perceptibility in a humanoid. The humanoid was tele-operated and successfully demonstrated capabilities to perform an ordinary task [120] in real world circumstances.

From 5+ years experience, the author suggests **a humanoid shared autonomy with a human controller** to fill the gap. While the long term goal is to make such humanoids fully autonomous, the tele-operational system is an emerging paradigm that has proved useful in allowing humans to control robots performing complex tasks. Human intelligence controls decision making and dictates commands to humanoids that handle low-level adaptation to environmental uncertainties. This dissertation demonstrates a design approach for **humanoids that fully function and complete a task in real world applications**.

3. BIPED LOCOMOTION: PLANNING AND STABILITY

Biped locomotion is a repeated motion which consists of three phases, the Right-Foot Single-Support Phase (RF-SSP), the Left-Foot Single-Support Phase (LF-SSP), and the Double-Support Phase (DSP). The bodies are not only moving forward but also swing side-to-side because of the phase transition. Such phase transitions also require the bodies accelerate or deaccelerate. For stable locomotion, the inertial forces produced by such dynamic movements of the bodies must counterbalance the gravitational forces within the support polygon. But, the support polygon is relatively very small and changed by the feet locations. This makes planning a reliable and stable locomotion challenging.

The approach widely used in humanoids basically requires 1) the desired ZMP trajectory, 2) a dynamic model of the humanoid, 3) stable CoM and foot trajectories designed from the ZMP trajectory via dynamic relationship, and 4) Inverse Kinematics (IK) to convert CoM and foot Cartesian coordinates to the humanoid's joint configuration. In this chapter, stable biped locomotion is demonstrated. The stability criterion, dynamic model, and gait planning algorithms for dynamic locomotion are reviewed. Minimum impact foot motion and analytic Inverse Kinematics (IK) are designed. Dynamic locomotion is fully simulated and experimentally evaluated.

3.1 Zero-Moment Point

In 1969, Vukobratovic et. al. [111][110] derived a mathematical relationship between the ankle torque and reaction force on the support foot for humanoid stability, called as the Zero-Moment Point (ZMP). The definition of ZMP is a point on the ground where the equivalent moment acting on the foot is zero. In other words, the

humanoid will not tip over as long as the ZMP lies within the area of the support polygon. It has been most widely used in humanoid robotics as a stability criterion.

Humanoids are supported by feet. The foot contact generates the reaction force that results in moment about the foot. This reaction moment induces a reaction torque at the ankle. The Zero-Moment Point (ZMP) is simply a dynamic relationship between the ankle and the reaction force and is used most widely as a humanoid stability criterion. The definition of ZMP originally explained in [111][110] is,

> *a point on the ground at which the net moment of the inertial forces and*
>
> *the gravity forces has no component along the horizontal axes.*

This idea states that there exists a point on the ground where the forces acting on the robot are dynamically balanced. In other words, the ZMP is located at the same point of CoM projected to the ground in a static case and otherwise they are not necessarily identical to each other.

Figure 3.1 shows a foot with ankle. The dynamic effect from the motion of the humanoid is simplified as a net force, F_{ankle}, and torque, M_{ankle}, acting on the ankle. The gravitational inertial force of the foot is denoted as $m_f g$ and there is a point, P_{react}, where the reaction force, F_{react}, and moment, M_{react}, exerted.

Let M_{ankle}, F_{ankle}, M_{react}, and F_{react} be decomposed into X, Y, Z components in Cartesian coordinates. They are expressed as (M_{ax}, M_{ay}, M_{az}), (F_{ax}, F_{ay}, F_{az}), (M_{rx}, M_{ry}, M_{rz}), and (F_{rx}, F_{ry}, F_{rz}), respectively. The horizontal reaction force, F_{rx} and F_{ry}, compensates F_{ax} and F_{ay}, respectively. If the surface condition is no slipping, then the horizontal reaction force is equivalent to the horizontal friction force and the reaction moment in Z axis, M_{rz}, is also balanced by that fiction force. The horizontal moment at ankle must be compensated by the moment induced by the vertical reaction force.
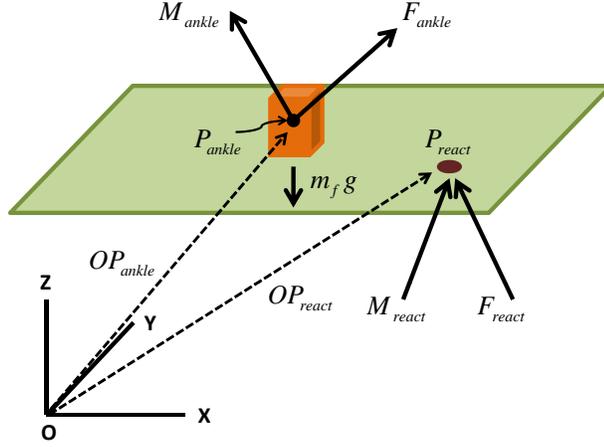
Figure 3.1: Free-body diagram of foot in the static equilibrium

$$M_{ax} = F_r z \times (P_{ry} - P_{ay}) \tag{3.1}$$

$$M_{ay} = F_r z \times (P_{rx} - P_{ax}) \tag{3.2}$$

where the position of P_{ankle} is described as (P_{ax}, P_{ay}, P_{az}). Also, the location of P_{react} is expressed as (P_{rx}, P_{ry}, P_{rz}).

Thus, if there exists a point, P_{react}, that satisfies Eq (3.1) and Eq (3.2), the reaction moment in the horizontal axes on that point will be,

$$M_{ax} = 0 \tag{3.3}$$

$$M_{ay} = 0 \tag{3.4}$$

Then, that point is called the Zero-Moment Point.

However, the size of foot is limited and may result in a ZMP location outside of the foot area. In this case, the reaction force acts on the edge of the foot and the

resultant moment is not zero at that point and results in the humanoid tipping over respect to that point.

The Zero-Moment Point can be easily calculated through static equilibrium.

$$F_{react} + F_{ankle} + m_f g = 0 \tag{3.5}$$

$$O\vec{P}_{react} \times \vec{F}_{react} + \vec{OG} \times m_f g + M_{ankle} + M_{rz} + O\vec{P}_{ankle} \times F_{ankle} = 0 \tag{3.6}$$

where $\vec{OG}$ is denoted as a vector from the origin in global frame to the center of gravity of foot, and $m_f g$ is the gravitational inertia of the foot.

If the origin in the global frame is coincident with the center of the convex hull in local frame and the mass of foot, m_f, and reaction moment in the vertical direction, M_{rz}, are ignored, the ZMP can simply be derived based on Eq (3.6).

$$F_{react} \times ZMP = -M_{ankle} \tag{3.7}$$

Humanoids consist of many rigid bodies connected to each other with a rotational joint. The definition of Zero-Moment Point, Eq (3.6), in Section 3.1 assumes that M_{ankle} and F_{ankle} is a net moment and force at the ankle, respectively, induced by the motion of the humanoid. To generalize the ZMP equation for a humanoid, the dynamics of a n-link humanoid using Newton and Euler equations replaces M_{ankle} and F_{ankle}.

$$f_a = \dot{P} - \begin{bmatrix} 0 & 0 & -m_t g \end{bmatrix}^T \tag{3.8}$$

$$\tau_a = \dot{L} - r_{com} \times \begin{bmatrix} 0 & 0 & -m_t g \end{bmatrix}^T \tag{3.9}$$

where f_a and τ_a are the 3×1 force and 3×1 torque vectors acting on the ankle, respectively. m_t stands for the total mass and g is the acceleration of gravity. r_{com} is

a position vector to the center of mass. $\dot{P}$ and $\dot{L}$ are the 3×1 matrices that denotes the change of linear and angular momentum defined in Eq (3.10) and Eq (3.11), respectively.

$$P = \sum_{i=1}^{n} m_i \dot{r}_i \tag{3.10}$$

$$L = \sum_{i=1}^{n} (r_i \times m_i \dot{r}_i + RI_i R^T w_i) \tag{3.11}$$

where w and r stand for the angular velocity and position of particle, and R is 3×3 rotational matrix.

By the definition of ZMP, the X and Y components in f_a are compensated by the horizontal reaction force and then only the force in the Z direction remains in f_a. Now, f_a and τ_a replace F_{react} and M_{ankle} in Eq (3.7) and the Z_{zmp} is assumed to be zero. Then, X_{zmp} and Y_{zmp} can be obtained by Eq (3.12) and Eq (3.13).

$$X_{zmp} = \frac{m_t g r_x + \dot{L}_y}{m_t g + \dot{P}_z} \tag{3.12}$$

$$Y_{zmp} = \frac{m_t g r_y + \dot{L}_x}{m_t g + \dot{P}_z} \tag{3.13}$$

where $\dot{L}_y$ and $\dot{L}_x$ denote the angular momentum about the Y- and X-axis. $\dot{P}_z$ is the linear momentum in the vertical direction.

Eq (3.12) and Eq (3.13) are the general form of humanoid's ZMP equation.

3.2   Simplified Dynamics: Linearized Inverted Pendulum

As long as the motion of a humanoid satisfies the Zero-Moment Point criterion, it could be said that such motion is dynamically or statically stable. Based on this idea, any stable walking motion can be designed with a mathematical model that
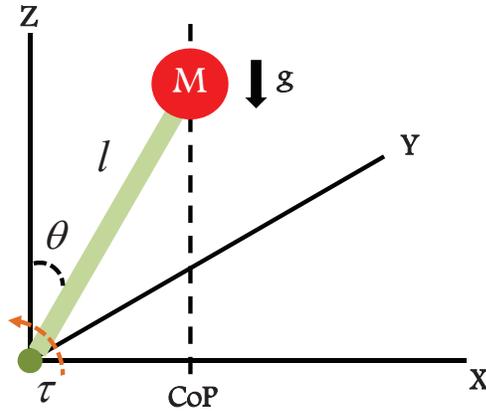
Figure 3.2: A simplified humanoid dynamic model: Three-Dimensional Linearized Inverted Pendulum Model

represents the dynamics of the humanoid.

More specific dynamic equations produce better walking motions and stability, but require more processing power. A simple linear inverted pendulum is employed based on the ZMP criterion to minimize the computational burden. The Three Dimensional Linearized Inverted Pendulum Model (3D-LIPM) [49] is a simplified dynamic model shown in Figure 3.2. XYZ is the global Cartesian coordinate system and M is a point mass. The resulting equations of motion of the pendulum model are non-linear, but can be linearized by assuming a massless bar, l, and constant height via small-angle approximation as shown below.

$$ml^2\ddot{\theta} = mgl\theta - \tau_y \tag{3.14}$$

where g is the gravitational constant and τ_y is the torque about the Y-axis. Assuming that the height of the COM remains constant, let $z_c = l$ where $l\theta$ is the X

position of COM. Eq (3.14) then becomes

$$\frac{z_c}{g}\ddot{x} = x - \frac{\tau_y}{mg} \tag{3.15}$$

The relationship between the X- and Y-directions for the COM (x_{com}, y_{com}) and the ZMP (x_{zmp}, y_{zmp}) are given by the ZMP definition defined in Eq (3.6) and Eq (3.7). The ZMP equation based on the 3D-LIPM in the X- and Y-directions can be obtained as

$$x_{zmp} = x_{com} - \frac{z_c}{g}\ddot{x}_{com} \tag{3.16}$$

$$y_{zmp} = y_{com} - \frac{z_c}{g}\ddot{y}_{com} \tag{3.17}$$

Eq (3.16) and Eq (3.17) imply that the net moment at the pivot of the pendulum is zero, which means that the torque generated by the reaction force due to the acceleration of the point mass is the same as the torque generated by the gravitational acceleration of the point mass. The mathematical expression of such dynamic relation is shown in Eq (3.18) below.

$$\tau_{zmp} = mg(CoM - zmp) - mz_c(CoM) \tag{3.18}$$

3.3   Reference ZMP Trajectory

Eq (3.16) and Eq (3.17) are the mathematical relationships between the Zero-Moment Point and the linearized inverted pendulum. They are non-homogeneous second order differential equations with a ZMP input and CoM output. Based on the desired ZMP input, a reliable CoM position can be calculated.

According to the ZMP stability criterion, the humanoid is balanced as long as the ZMP is located within the support polygon. The support polygon is determined by
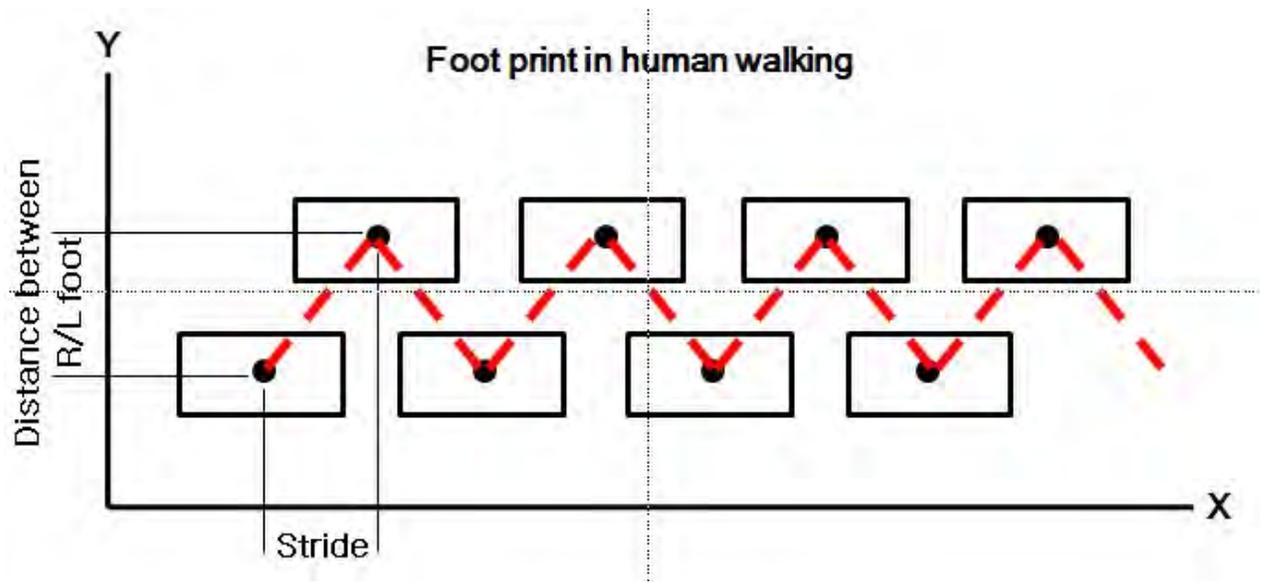
Figure 3.3: Zero-Moment Point in each support foot and its desired trajectory in walking

the walking sequence that basically consists of a Single-Support Phase (SSP) and a Double-Support Phase (DSP). With a proper selection for ZMP in walking phases, the ZMP trajectory can be obtained. Figure 3.3 shows a typical foot print during forward walking with an arbitrary stride. Let the ZMP be located in the center of the convex hull and connect the ZMP to next ZMP by a line. Then, that line becomes the ZMP trajectory in X and Y coordinate system.

However, walking is a dynamic motion which is a function of time. A cycle of walking consists of SSP and DSP. Thus, the proper SSP and DSP periods must be specified. In static walking, the ratio of the SSP period to DSP period is not considerable but critical in dynamic walking[1]. Once SSP and DSP periods are suitably determined, the desired ZMP trajectories in the forward, X, and lateral, Y, directions are specified. Figure 3.4 depicts the typical ZMP trajectory in X and Y direction

---

[1]According to a formulation in Kim et. al. [59], the SSP period takes 95% of a walking cycle and the rest is the DSP period. On the other hand, based on the author's observation of human walking, the SSP period is a range of 15 to 20% of a walking cycle. In fact, there is no standard period for both phases and it has to be optimized via experiments.
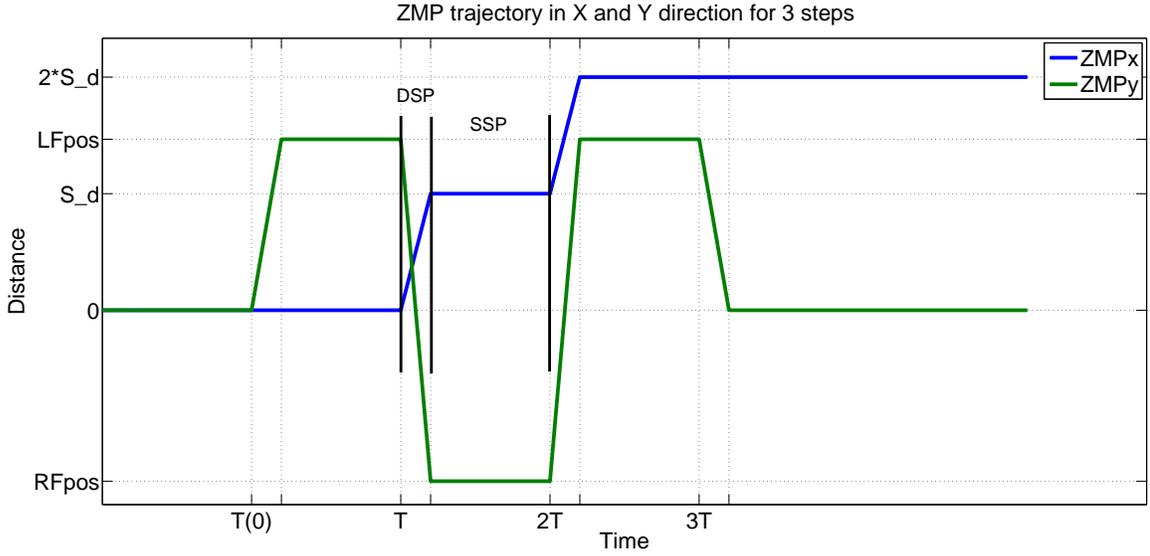
Figure 3.4: ZMP trajectory for wakling forward in X and Y direction with DSP and SSP period. *LFpos* is a desired left foot position in Y for each step and *RFpos* means accordingly.

for 2 steps walking forward. *LFpos* and *RFpos* are left foot and right foot position, respectively, and S_d denotes the forward stride. $T(0)$ is the time at rest and T is the walking period for each step.

3.4 Numerical Approach: Optimal ZMP Preview Control

The ZMP equation is derived based on 3D-LIPM in Section 3.2. Now we need to generate a reliable CoM trajectory from the ZMP equation. The optimal ZMP preview controller is one of such methods to produce the CoM trajectory in X and Y direction, proposed by Kajita et. al. [47] with the idea of previewable optimal control published by Katayama et. al. [53]. This controller requires the future reference ZMP trajectory to track current reference ZMP trajectory without delay.

The optimal ZMP preview control defined a new control input to represent ZMP equations as a state-space. The new input is the time derivative of the horizontal

acceleration of the center of mass, defined in Eq (3.19).

$$\frac{d}{dt}\ddot{P}_{com} = u \tag{3.19}$$

P_{CoM} denotes the position of the center of mass, (X_{com}, Y_{com}), and u is the new control input. Now, the ZMP equation in the X direction can be transformed to state-space representation in Eq (3.20) and Eq (3.21).

$$\frac{d}{dt}\begin{bmatrix} X_{com} \\ \dot{X}_{com} \\ \ddot{X}_{com} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_{com} \\ \dot{X}_{com} \\ \ddot{X}_{com} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u_x \tag{3.20}$$

$$X_{zmp} = \begin{bmatrix} 1 & 0 & -\frac{z_c}{g} \end{bmatrix} \begin{bmatrix} X_{com} \\ \dot{X}_{com} \\ \ddot{X}_{com} \end{bmatrix} \tag{3.21}$$

The state-space representation for the Y direction can be transformed in the same manner as Eq (3.20) and Eq (3.21). Now Eq (3.20) and Eq (3.21) need to be discretized as follow.

$$\begin{aligned} X(k+1) &= AX(k) + Bu(k) \\ P(k) &= CX(k) \end{aligned} \tag{3.22}$$

Where

$$X(k) = \begin{bmatrix} X_{com}(kT) & \dot{X}_{com}(kT) & \ddot{X}_{com}(kT) \end{bmatrix}^T$$

$$u(k) = u_x(kT)$$

$$P(k) = X_{zmp}(kT)$$
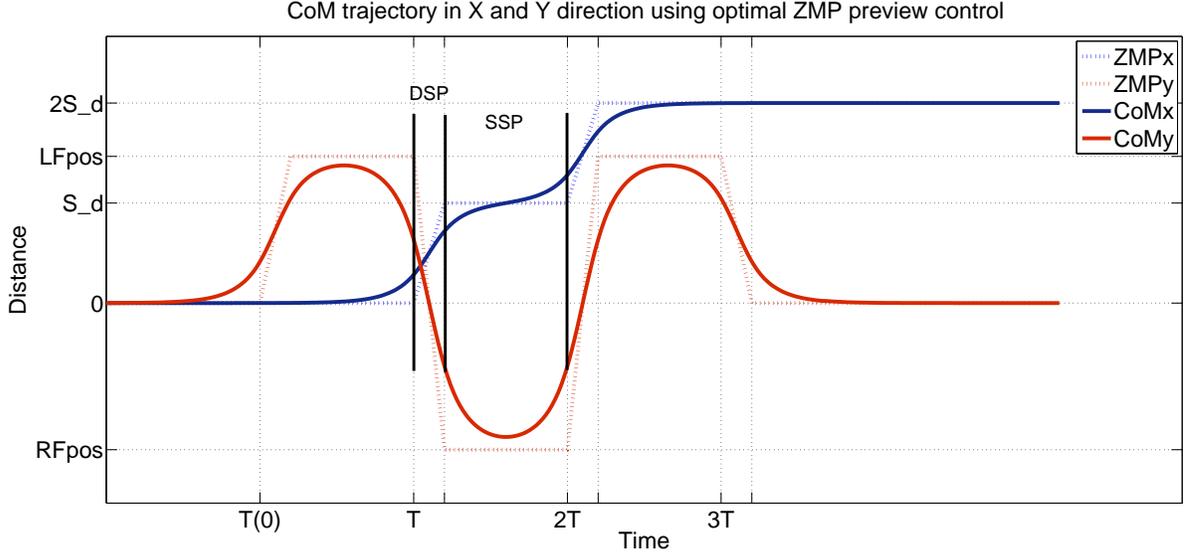
Figure 3.5: CoM trajectories based on a desired ZMP trajectory (dotted) for X and Y direction. S_d denotes the step distance in the forward direction, and $LFpos$ and $RFpos$ are the left and right foot position in each step, respectively. T is the walking period for each step.

$$A = \begin{bmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{T^3}{6} \\ \frac{T^2}{2} \\ T \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & -\frac{z_c}{g} \end{bmatrix}$$

T is the sampling time. The discrete-time state space model for the lateral direction can be realized in the same manner.

To minimize the tracking error $e(i) = p(i) - p^{ref}(i)$ and be asymptotically stable,

the integral action is required in the formulation [53]. Let $\Delta X(k) = X(k) - X(k-1)$ be the incremental state vector and $\Delta u(k) = u(k) - u(k-1)$ be the incremental input. The performance index that needs to be minimized is defined as

$$J = \sum_{i=k}^{\infty} Q_e e(i)^2 + \Delta X^T(i) Q_x \Delta X(i) + R \Delta u^2(i) \tag{3.23}$$

where $e(i) = p(i) - p^{ref}(i)$ is the ZMP error, Q_e, $R > 0$ and Q_x is a 3×3 symmetric semi-definite matrix.

For the integral action, the augmented state is

$$\tilde{X} = \begin{bmatrix} e(k) \\ \Delta X(k) \end{bmatrix} \tag{3.24}$$

Then,

$$\tilde{A} = \begin{bmatrix} I & CA \\ 0 & A \end{bmatrix} \tag{3.25}$$

$$\tilde{B} = \begin{bmatrix} CB \\ B \end{bmatrix} \tag{3.26}$$

$$\tilde{C} = \begin{bmatrix} I & 0 & 0 & 0 \end{bmatrix} \tag{3.27}$$

The optimal controller to minimize Eq (3.23) is given by Kajita et. al. [47].

$$u(k) = -G_i \sum_{i=0}^{k} e(k) - G_x X(k) - \sum_{j=1}^{N_L} G_p(j) p^{ref}(k+j) \tag{3.28}$$

Where G_i and G_x are the optimal gains of the system calculated by Discrete Algebraic Riccati Equation (DARE) and and G_p is an optimal gain for previewed
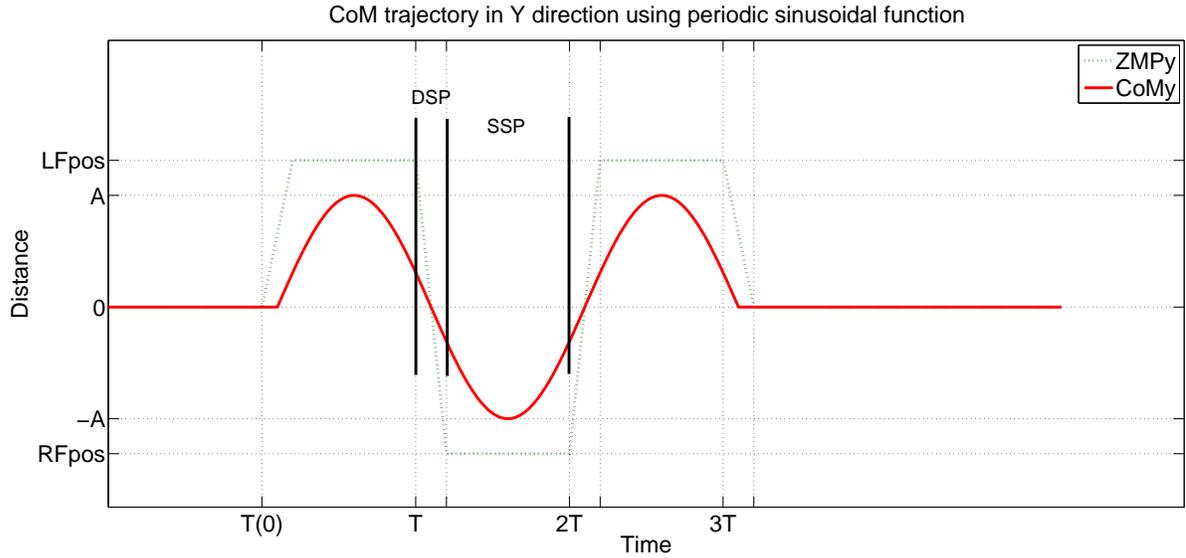
Figure 3.6: CoM trajectory in the Y direction using the periodic sinusoidal function based on a desired ZMP trajectory (dotted) in Y direction. *LFpos* and *RFpos* are left and right foot position in each step, respectively. T is the walking period for each step.

inputs that can be obtained through iterative manner [47][90]. N_L is the number of future inputs which contributes the future ZMP reference trajectory.

The size of N_L empirically determined is more than $1.5sec \times frequency$ [47]. The decision of the weight, Q_e, Q_x, and R, is straightforward. A set of values for them are $Q_e = 1.0$, $Q_x = 0$, and $R = 1.0 \times 10^{-6}$ [47].

Based on the desired ZMP trajectory, a reliable CoM trajectory can be generated, and shown as Figure 3.5.

3.5   Analytic Approach: Periodic Sine-Cycloid Planning

The periodic sine-cycloid walking pattern generation method, proposed by Kim et. al. [59], is a method used to plan a stable CoM trajectory along the sagittal and coronal axes. It assumes that the walking is a periodic motion and can be designed as
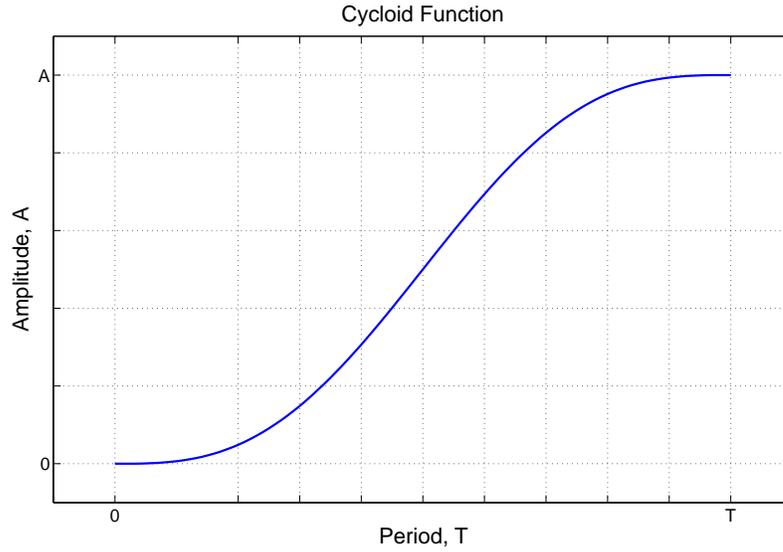
Figure 3.7: Cycloid function: $2\pi t - sin(2\pi t)$

a sinusoidal function. Based on a simplified dynamic model of the linearized inverted pendulum, the CoM motion follows a sinusoidal trajectory along the coronal axis as Eq (3.29).

$$Y_{com} = A \sin wt \tag{3.29}$$

where Y_{com} is the CoM position of the humanoid in lateral direction, A represents the amplitude of the sway of the humanoid's CoM along the Y axis, and w denotes the walking frequency which is $2\pi f$ where f is the natural frequency of the inverted pendulum model.

To calculate the ZMP in the Y direction, $\ddot{Y}_{com}$ needs to be derived as shown in Eq (3.31) derived from Eq (3.30).

$$\dot{Y}_{com} = Aw \cos wt \tag{3.30}$$

$$\ddot{Y}_{com} = -Aw^2 \sin wt \tag{3.31}$$
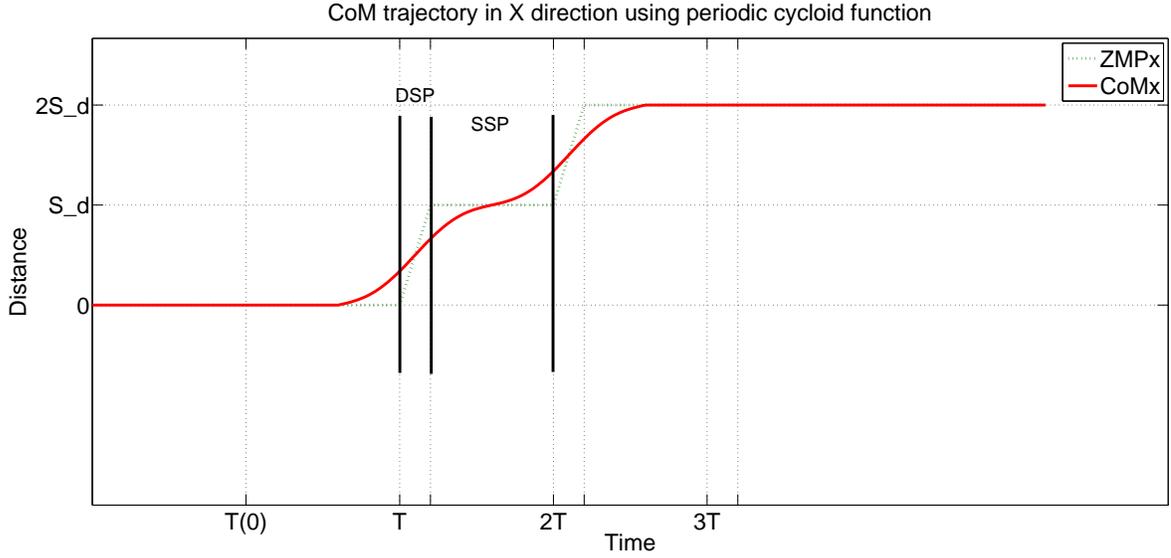
Figure 3.8: CoM trajectory in X direction using the periodic cycloid function based on the desired ZMP trajectory (dotted) in X direction. S_d denotes the forward step distance and T is the walking period.

Combining Eq (3.29) and Eq (3.31) with Eq (3.17), the Y_{zmp} is re-defined in Eq (3.32).

$$Y_{zmp} = A(1 + \frac{z_c}{g}w^2)\sin wt \tag{3.32}$$

According to Section 3.1, the ZMP is designed to be placed in the center of the convex hull. In other words, it is located at the center of the support foot in Single-Support Phase (SSP). Hence, the desired ZMP in the Y direction is the same as the lateral displacement of each foot shown in Figure 3.4. As a result, the amplitude, $A(1 + \frac{z_c}{g}w^2)$, matches to the desired Y_{zmp}, and then the unknown parameter, A, can be calculated as Eq (3.33).

$$A = \frac{Y_{zmp}}{1 + \frac{z_c}{g}w^2} \tag{3.33}$$

To complete Eq (3.33), the walking frequency, $w = 2\pi f$, must be defined. Based on Eq (3.14) from Figure 3.2, the natural frequency of the inverted pendulum is easily derived as Eq (3.34).

$$f = \frac{1}{2\pi}\sqrt{\frac{g}{z_c}} \tag{3.34}$$

With a given CoM height, z_c, the natural frequency, f, is specified and it leads to the optimal SSP period, $\frac{1}{2f}$. It also yields for A from Eq (3.33). However, one must note that the parameter, A, should be tuned for practical implementation because of modeling uncertainties and mechanical compliance that could exist. The desired CoM trajectory in the Y direction is shown in Figure 3.6.

To find the CoM motion in the forward direction, the periodic cycloid function is employed. A cycloid is a curve defined by the path of a point on the edge of a rolling wheel (Figure 3.7).

$X_{com}(t)$ represents this path over time,

$$X_{com} = \frac{B}{2\pi}(2\pi ft - sin(2\pi ft)) \tag{3.35}$$

where B is the amplitude and f is the frequency defined as $\frac{1}{T}$ when T is the walking period which is the sum of the SSP and DSP.

In order to combine Eq (3.35) with the ZMP equation in Eq (3.16) for the X-direction, the second derivative of the cycloid function (Eq (3.36)) is needed.

$$\ddot{X}_{com} = \frac{B}{2\pi}((2\pi f)^2 sin(2\pi ft)) \tag{3.36}$$

Eq (3.35) and Eq (3.36) are combined

$$X_{zmp} = \frac{B}{W2\pi}(W2\pi ft - (1 + \frac{z_c}{g}(2\pi f)^2 sin(2\pi ft))) \tag{3.37}$$

where W is the weighted value to control the acceleration of the hip in the double-support phase (DSP), B is the step distance, and f is $\frac{1}{T}$ when T equals the sum of the DSP and SSP.

Figure 3.8 shows the hip trajectory in the X-direction using a cycloid algorithm for a given ZMP trajectory.

3.6 Minimum Landing Impact Foot Motion

Humanoid feet are the only body part that contact the ground when walking and hence play an important role in balance. The humanoid consists of many rigid bodies and mostly does not have any mechanical force compliance. This results in ab impulsive contact force during foot landing. To minimize such contact reaction forces, the foot has to deaccelerate before contact. Such trajectories can be designed using a third-order interpolation or cycloid function described in Section 3.5. The author designs the foot motion using the cycloid function which deaccelerates its motion before contacting the ground [41].

The motion of the foot is designed in Cartesian space that consists of 3 components in the X, Y, and Z directions. To walk forward in biped motion, one foot must swing in front of the other. The ground friction prevents the planted foot from slipping and allows the motion to carry through.

The foot motion fundamentally consists of three phases; lifting, swing, and landing. Figure 3.9 depicts the foot-motion sequence of a human. Based on Figure 3.9, to design the foot motion requires two main parameters; stride, and lifting height. In addition, such foot motions begin and end during the Single-Support Phase (SSP), composed of the Upward Single-Support Phase (USSP) and Downward Single-Support Phase (DSSP).

In terms of parameters, the strides in the X and Y directions are obtained from
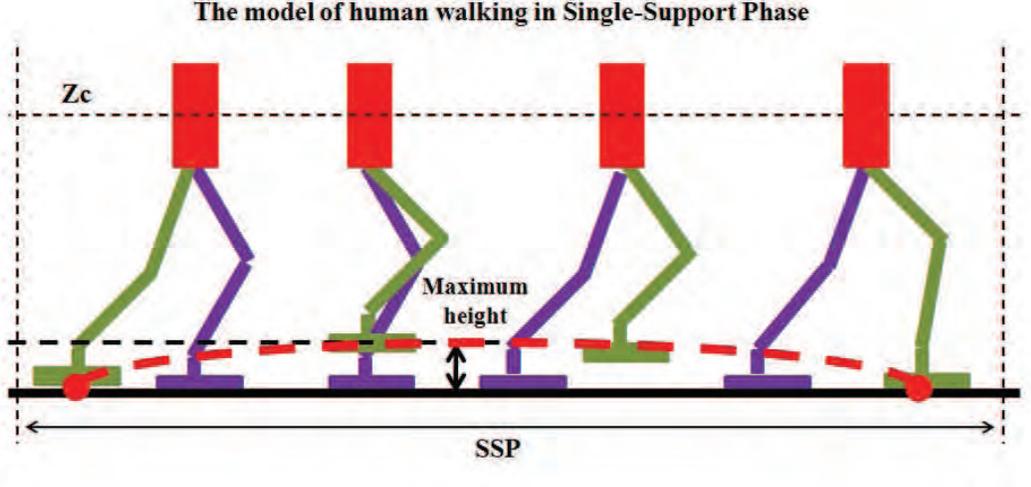
Figure 3.9: Human foot motion in a walking step. It consists of lifting, swing, and landing motion of foot.

the ZMP trajectory that is a trace of the center of the convex hull and vise-versa. The lifting height for the horizontal movement of foot is actually an arbitrary value specified manually. With those parameters, the foot trajectory in Cartesian space can be designed. Eq (3.38) and Eq (3.39) use a third-order interpolation method, a cycloid function, to generate the foot trajectories in the X and Y directions, respectively.

$$Foot_x = \frac{S_d}{2\pi}(2wt_{SSP} - sin(2wt_{SSP})) \tag{3.38}$$

$$Foot_y = \frac{S_y d}{2\pi}(2wt_{SSP} - sin(2wt_{SSP})) \tag{3.39}$$

where S_{xd} and S_{yd} are the step distance in the X and Y directions. w denotes $2\pi f$, $f = \frac{1}{t_{SSP}}$, and t_{SSP} is the SSP duration that equals to the sum of the $USSP$ and $DSSP$ periods.

For the Z direction, Eq (3.40) and Eq (3.41) yield foot trajectories in the Z-directions when $0 \leq t < \frac{USSP}{2}$ and $\frac{USSP}{2} < t \leq DSSP$, respectively.
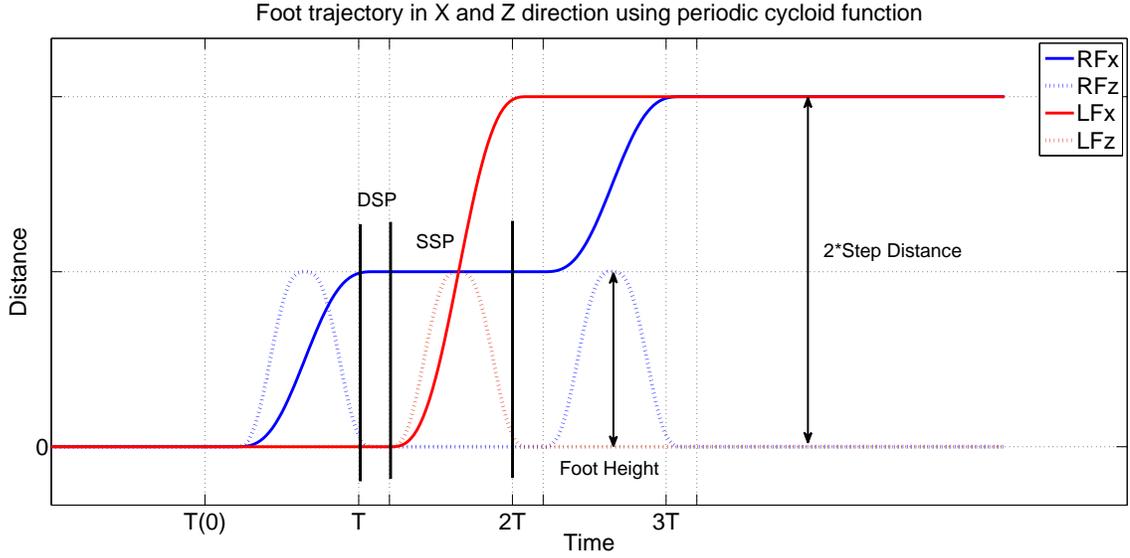
Figure 3.10: Right and left foot trajectories in the X and Z directions with a walking stride in the forward direction, X, and lifting height described as $FootHeight$. T is a walking cycle including the DSP and SSP. The dotted lines are the foot motion in the Z direction and the bold lines are in X direction.

$$Foot_z = \frac{H}{2\pi}(2wt - sin(2wt)) \tag{3.40}$$

$$Foot_z = 2H + \frac{H}{2\pi}(sin(2wt) - 2wt) \tag{3.41}$$

where H is the maximum height of the foot motion height.

Eq (3.38), and Eq (3.40) and Eq (3.41) yield plots given in Figure 3.10 for step distance, S_d, and maximum foot height H.

Note that the cycloid function method decelerates the foot motion at landing shown in a plot for motion in the Z direction in Figure 3.10. Such trajectories minimize the impulsive contact reaction force at landing.

3.7   Analytic Vector Inverse Kinematics

Inverse Kinematics is a mathematic strategy to map the position of an end-effector in Cartesian space to the joint angles in the joint configuration space. Each rigid body in a humanoid is connected by a rotational joint to one or more adjacent rigid bodies. Regardless of the types of joints like a linear or rotational joint, the motion of the humanoids can be achieved through setting desired joint angles. Motion planning in joint space with a high Degree Of Freedom (DOF) is very implicit and ambiguous since there are many possible solutions for any given end-effector state. Therefore, the typical way to solve this ambiguity is to design a motion in Cartesian space and then to convert it to joint angles while imposing constraints.

There are two ways to solve IK. The first method is to calculate the joint angles analytically. This is the explicit approach based on the math that gives an exact solution [5]. However, it has a critical problem when either many solutions (redundant joint) or no solutions (unreachable end-effector state) exist. In these cases, the IK solver returns an error that may cause the system to fail. To avoid these cases, many kinematic constraints must be applied to the IK solver but these could result in a non-natural motion or non-smooth joint trajectory.

The other method is to find the solution numerically [9]. This strategy solves the IK problem based on iterative methods with an optimization algorithm. Therefore it can avoid mathematical singularities and is appropriate for a high DOF system. It does not need kinematic constraints but is computationally expensive and subject to homing in on local mimimas. Such limitations are still an open issue in the field of IK.

Humanoids typically have more than 30 DOF. Solving all joint angles at once using an IK algorithm can be very computationally expensive and yield singularities. Additionally, slow computation often misses samples of sensory data collected during
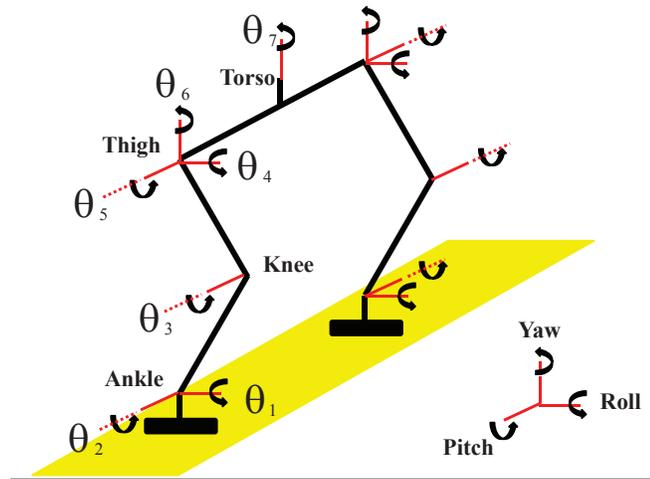
Figure 3.11: Typical joint configuration (13 DOF) for the low-body in a humanoid

processing. This phenomena is critical in real-time control systems. Missing samples often yield undesired responses in controllers. The author derive an analytic IK solver in vector space with some kinematic constraints [41].

As depicted in Figure 3.11, a pair of humanoid legs typically consist of at least 13 degrees-of-freedom (DOF). Pattern generators typically solve the inverse kinematics (IK) for each leg (six DOF) and treat the hip as an end effector and the foot as the origin. Computational costs often prevent direct control of hip position.

One casually observes that people typically do not yaw their torso when walking straight. One explanation is perhaps the conserve angular momentum except when turning or intentionally taking long strides. Taking advantage of this zero yaw observation prescribes four constraints on the thighs:

1. Angular momentum from yaw is zero

2. Upper body is always perpendicular to the ground

3. Bottom of foot is always parallel to the ground
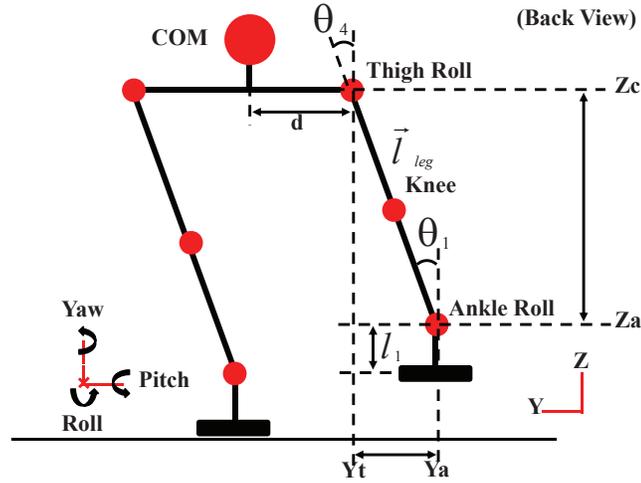
Figure 3.12: Vector-space inverse kinematics calculation for roll joints in back-plane view

4. Height of COM is constant

With these constraints, the three pitch angles and two roll axes in each leg can be analytically calculated in the vector-space. This is important because these constraints allow real-time hip-position control without demanding extensive processing power.

Figure 3.12 shows the roll angles for the ankle, θ_1, and thigh, θ_4, in the back-plane of the humanoid. Z_a is the ankle height and is equal to $Z_f + l_1$, where Z_f is the foot height and l_1 is the ankle-to-foot length. The Y positions for the thigh, Y_t, and hip, Y_h, are related by $Y_t = Y_h - d$. $\vec{l_{leg}}$ denotes the 2D vector between the thigh and ankle joints.

Through trigonometry θ_1 and θ_4 can be determined as

$$\theta_1 = 90 - \arccos((Y_t - Y_a)/|\vec{l_{leg}}|) \tag{3.42}$$
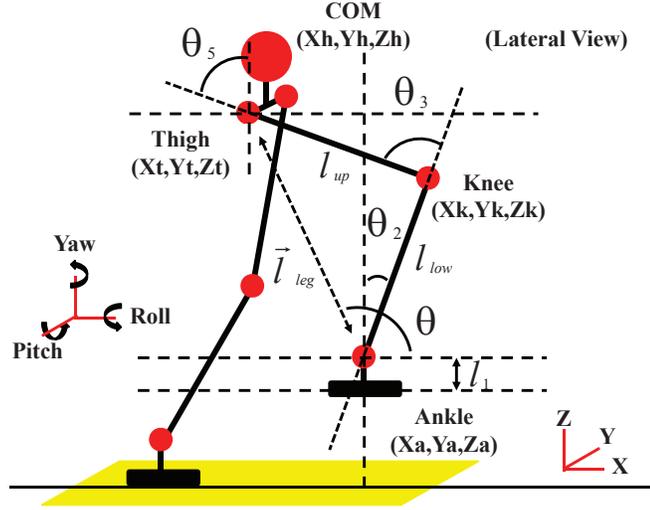
Figure 3.13: Vector-space inverse kinematics calculation for pitch joints in lateral-plane view

$$\theta_4 = -\theta_1 \tag{3.43}$$

where

$$|\vec{l_{leg}}| = \sqrt{(Z_c - Z_a)^2 + (Y_t - Y_a)^2} \tag{3.44}$$

All pitch angles can be calculated using the same approach. Figure 3.13 shows the ankle, θ_2, knee, θ_3, and thigh, θ_5, pitch angles. Let X, Y and Z symbols respectively denote joint positions in the X-, Y-, and Z-directions.

Furthermore, let the upper and lower leg link lengths be respectively denoted by l_{up} and l_{low}. $\vec{l_{leg}}$ is the 3D vector between the thigh and ankle. Applying the law of cosines yields

$$\theta_3 = 180 - \arccos((l_{up}^2 + l_{low}^2 - \vec{l_{leg}}^2)/(2l_{up}l_{low})) \tag{3.45}$$

$$\theta_5 = 90 - \theta - \arccos((l_{up}^2 + \vec{l_{leg}}^2 - l_{low}^2)/(2l_{up}|\vec{l_{leg}}|)) \tag{3.46}$$

$$\theta_2 = 90 - \theta + \arccos((l_{low}^2 + \vec{l_{leg}}^2 - l_{up}^2)/(2l_{low}|\vec{l_{leg}}|)) \tag{3.47}$$

where

$$|\vec{l_{leg}}| = \sqrt{(Z_t - Z_a)^2 + (Y_t - Y_a)^2 + X_t - X_a)^2} \tag{3.48}$$

$$\theta = \arccos((X_t - X_a)/|\vec{l_{leg}}|) \tag{3.49}$$

Those equations can be applied to the CoM trajectory for a horizontal and vertical motions given the foot trajectory to convert a Cartesian path to joint angles in lower body of the humanoid.

Note that the thigh yaw joint in this vector space calculation is assumed to be zero. The humanoid, however, can turn to change its moving direction. In this case, the thigh yaw joint angle must be solved. Since each joint angle is calculated independently in vector space, the thigh yaw joint can be obtained simply by the angle of turning. Regardless of types of locomotion such as walking and climbing, this vector IK solves all joint angles in the lower body without any additional computation and constraint.

In addition, to compare computational times, calculations were performed on a dual core 3.37 GHz Pentium 4 running MATLAB. Traditional Jacobian pseudo IK using damped least squares yielded joint angle calculations for 621 trajectory samples in 0.095 seconds. By contrast, calculation time was reduced more than 50% (0.042 seconds) when yaw motions were constrained to zero.

The vector IK solves the lower body angles. Now, the upper-body orientation is defined by roll, pitch, and yaw. Typical mechanical design of humanoids has independent joints for the upper-body yaw. But, these robot often do not have independent upper-body roll and pitch joints because of limited space in the design. Instead, the roll and pitch of the upper-body are usually coupled to the lower body

motion. Thus, the solution of vector IK for lower-body is determined by the value of upper-body orientation.

Let a 3×3 Euler angle set, R_e, be associated with roll θ_r, pitch θ_p, and yaw θ_y. The composition of R_e depends on the order of the applied transforms. Then, the orientation of the current upper-body posture, R_o, becomes $R_o = R_e$. Using Eq (3.45), Eq (3.47), and Eq (3.42), the angles for knee pitch θ_3, ankle pitch θ_2, and ankle roll θ_1 are calculated. Based on the forward kinematics, the 3×3 rotational matrix, R_f, is defined with θ_3, θ_2, and θ_1. Then, Eq (3.50) is valid.

$$R_e = R_f^T R_o \tag{3.50}$$

Based on Eq (3.50), the new thigh pitch θ_5 and thigh roll θ_4 are obtained easily and the newly defined thigh yaw, θ_6, can be computed simply.

3.8   Simulation and Experiment

The gait patterns for walking forward were generated by the Optimal ZMP Preview controller in Section 3.4 and simulated with the vitual-Hubo (see Appendix A) in the Webots simulation environment. Figure 3.14 is a snapshot of virtual-Hubo walking 20 steps forward. For this walking gait, the CoM height was set to $230mm$, the stride was $100mm$, and the foot height was $30mm$. The periods for the DSP and SSP were 0.2 seconds and 0.8 seconds, respectively. The trajectory was sampled every 0.2 milliseconds since the control frequency of virtual-Hubo was 50Hz. With given CoM and foot trajectories, the vector IK was applied to convert them into joint angles. These joint angles were executed in this simulation.

Figure 3.15 is another snapshot of virtual-Hubo walking 20 steps forward. This dynamic walking was planned via periodic sine-cycloid approach with the same parameters as used in optimal ZMP preview control. Both planning algorithms were

fully valid in simulation.

A full-sized humanoid called online-Hubo was used to verify and validate the gait patterns. Figure 3.16 is a snapshot of online-Hubo walking 2 steps in the forward direction. The biped locomotion was realized via optimal ZMP preview control with cycloid foot motion. The stride was $130mm$ and foot height was $100mm$. The SSP period was set to 16 seconds while the DSP period was 10 seconds. The gait was a quasi-static open-loop walking gait used only to validate the gait pattern.
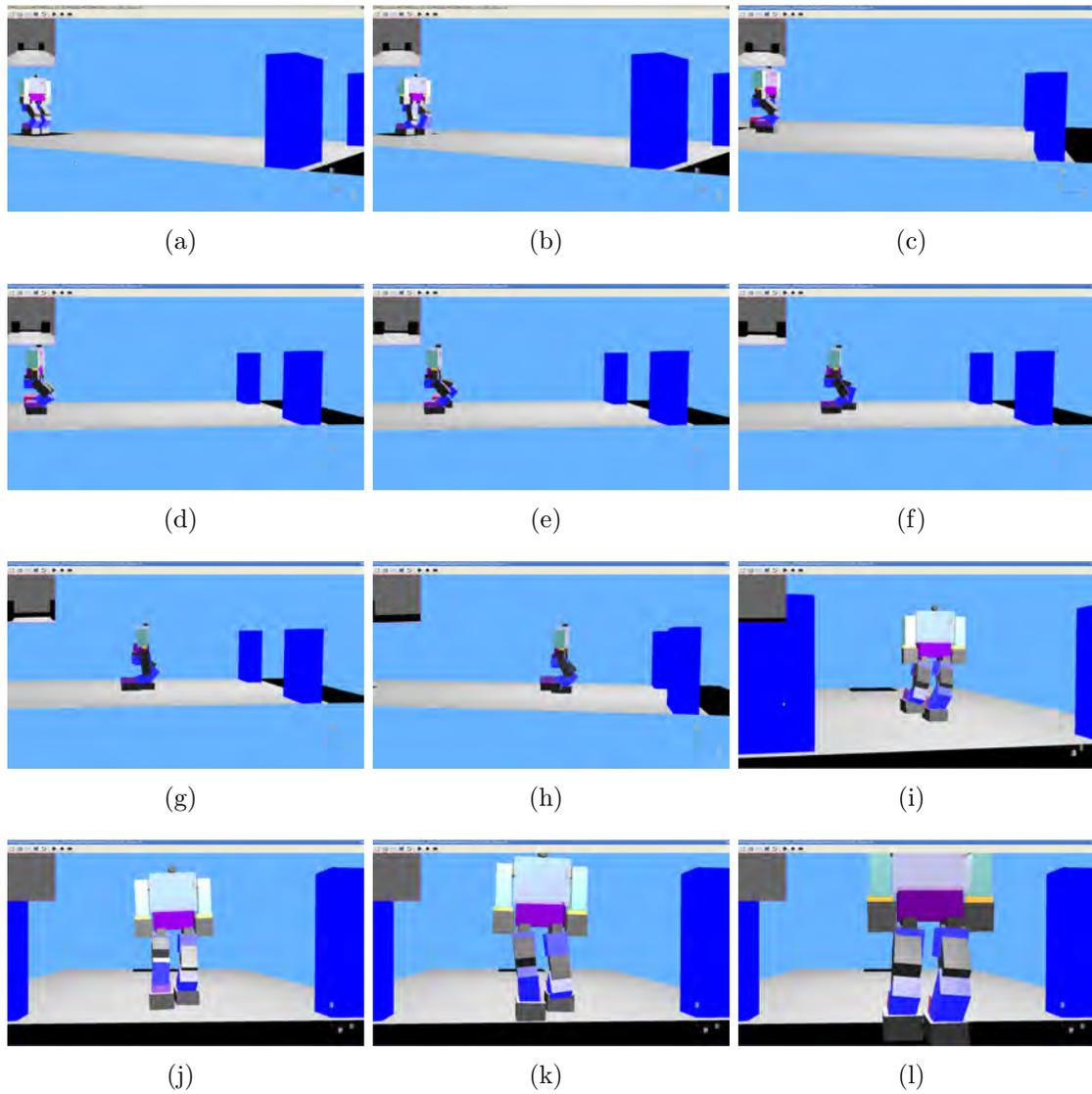
Figure 3.14: Optimal ZMP preview control simulation on virtual-Hubo with cycloid foot motion. The SSP and DSP periods were 0.8 and 0.2 seconds, respectively.

(a)   (b)   (c)

(d)   (e)   (f)

(g)   (h)   (i)

(j)   (k)   (l)
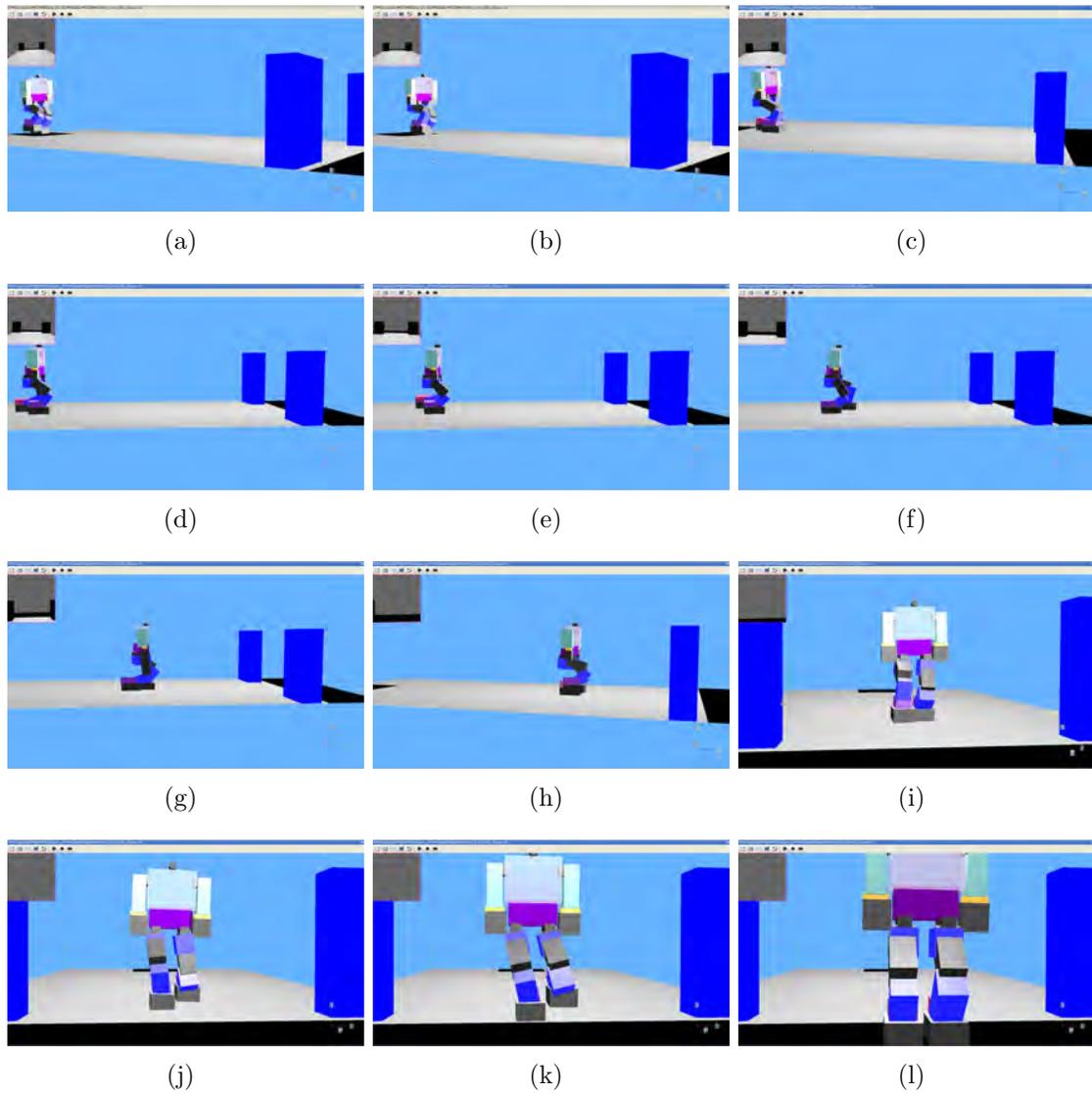
Figure 3.15: Periodic Sine-Cycloid algorithm simulation on virtual-Hubo with cycloid foot motion. The SSP and DSP periods were 0.8 and 0.2 seconds, respectively.
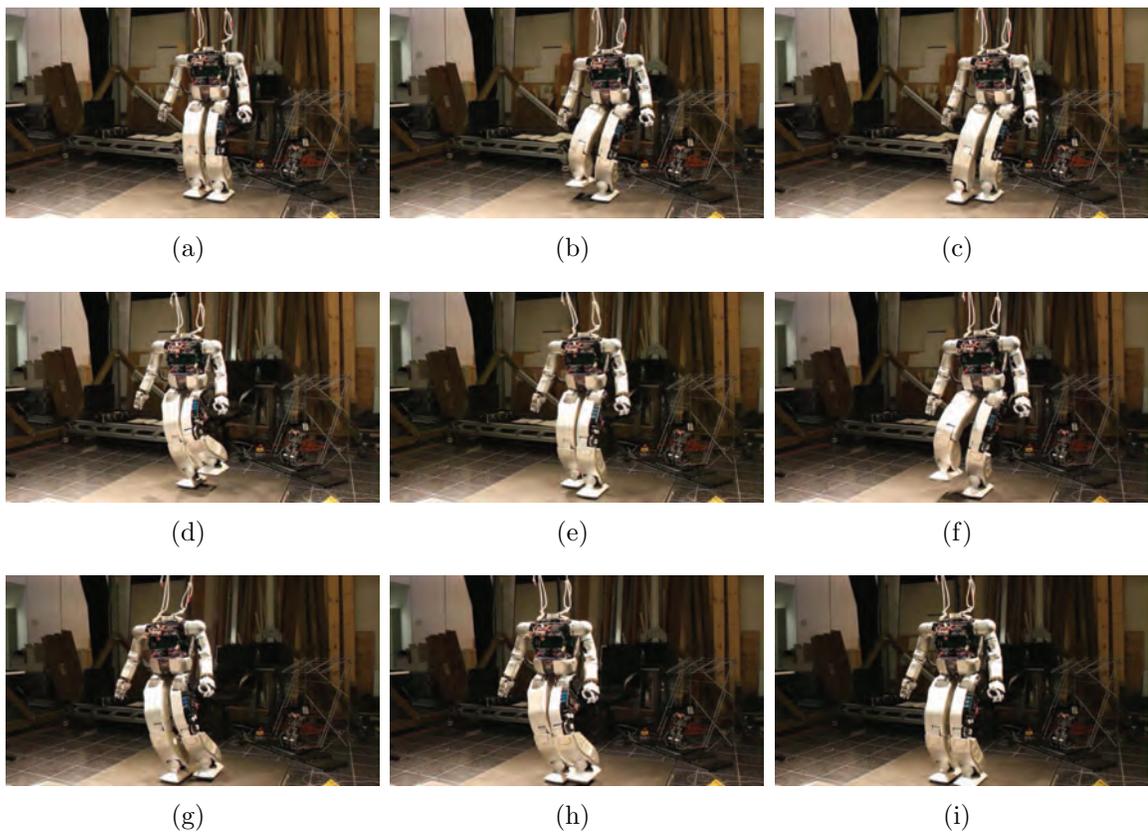
Figure 3.16: Online-Hubo walking 2 steps in the forward direction. Gait patterns are produced through optimal ZMP preview control with cycloid foot motion.

4. BIPED LOCOMOTION ON UNEVEN TERRAINS

Biped walking is a unique feature of humanoids. It gives humanoids mobility when performing tasks in human-centered environments like homes, offices, factories, and even unstructured surroundings. Current state-of-the-art humanoids have demonstrated dynamic walking on stairs [60][41], gravel-covered fields [8], and even lawns [96]. With an accurate sensory perception of environments, these robots are able to plan and control their dynamic gait to traverse a specific terrain. The long term goal of such works is to make these robots fully autonomous to travel all terrains that human beings are capable of traversing. These robots will some day partake in tasks like logging in the mountains, repairing a bridge spanning a small stream, and servicing a power line in inclement weather. However, the current state-of-the-art lacks the answers to how humanoids will tackle these challenges. The environments may be covered with water, snow, or brush and the ground may be obscured. Natural disturbances from wind and water impose forces on the humanoid when moving. These factors demand the ability for humanoids to handle such unknowns while walking to maneuver effectively.

In this chapter, stable and adaptable biped gait planning for locomoting unstructured terrains is demonstrated. The works show humanoids' abilities to avoid obstacles, follow a given path, climb stairs, and locomote over uneven terrains. All methods of locomotion are experimentally verified and validated.
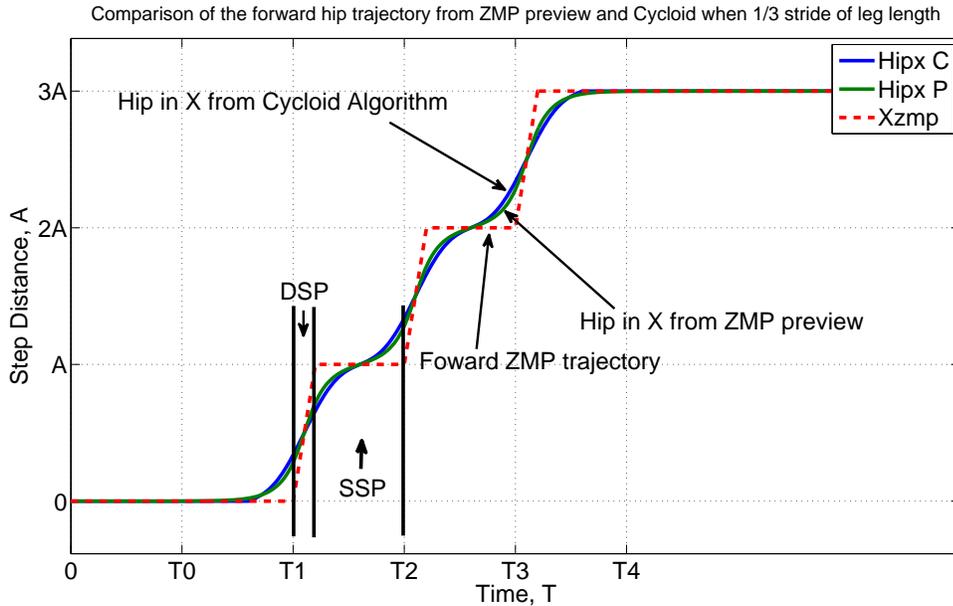
Figure 4.1: CoM trajectory comparison between optimal ZMP preview control (indicated by arrow) and the periodic cycloid function (indicated by arrow) for the X direction with the same walking period and step distance
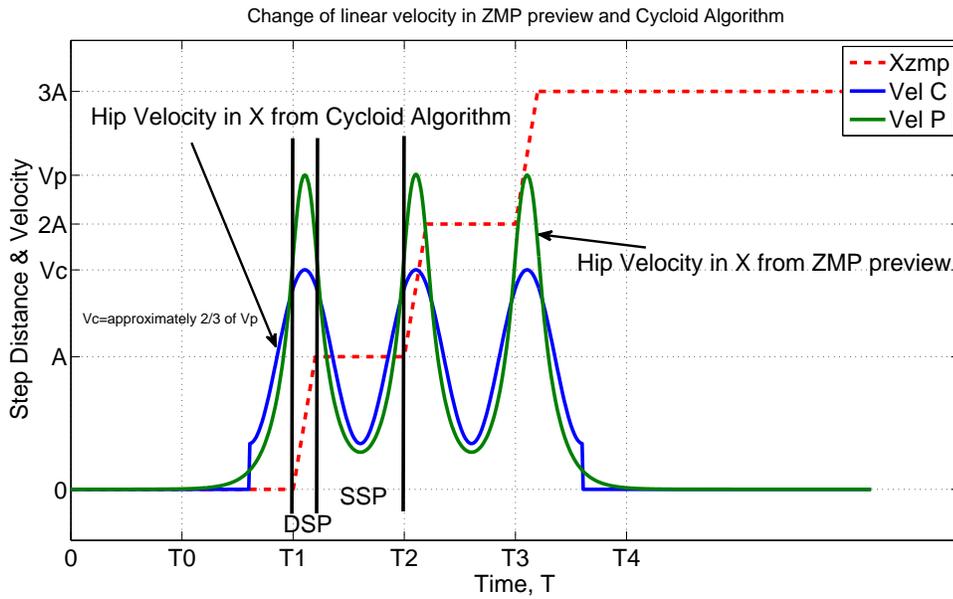
Figure 4.2: CoM linear momentum comparison between the trajectories shown in Figure 4.1

4.1   Real-Time Gait Planner Combined with Periodic Sine-Cycloid and Optimal ZMP Preview Control Methods

The optimal ZMP preview controller generates a reliable CoM trajectory automatically based on a given dynamic model and desired ZMP trajectory. However, the dynamic model is a mathematical model that cannot accurately represent the actual humanoid without error. Such modeling discrepancies result in an unreliable CoM trajectory. In this case, such trajectory has to be refined or tuned. Such tuning is not allowable when using the optimal ZMP preview controller without modifying the dynamic model.

Such modeling error is apparently observed in a walking gait that requires a relatively large stride in the forward direction; for example, the walking pattern for stair climbing or stepping over an obstacle. The optimal ZMP preview controller generates a forward CoM trajectory that has a large linear momentum. This results in the humanoid tipping over when stepping over a relatively long distance.

In contrast, the periodic sine-cycloid function algorithm actually designs the CoM trajectory in such a way that a possible periodic function fits into the desired ZMP trajectory. Thus, such modeling errors can be tuned. By changing the value of the weight function, the linear momentum can be adjusted. As a result, the vibrations in the walking phase transition associated with changing the support foot can be minimized.

In Figure 4.1, the CoM trajectories in the forward direction from the optimal ZMP preview controller and periodic cycloid function are compared. Figure 4.2 shows the linear momentum plots of both algorithms based on the same walking cycle and stride, set to one-third of the leg length. In Figure 4.2, the linear momentum of the forward trajectory using the periodic cycloid function is $\frac{2}{3}$ of that of the optimal ZMP preview control algorithm.
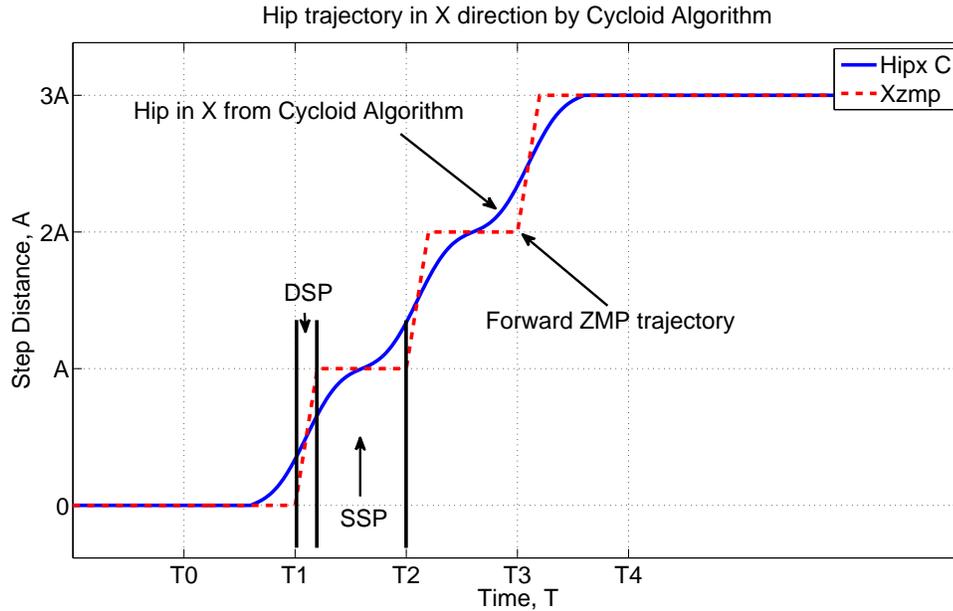
Figure 4.3: Dynamic walking gait for forward (x) walking generated by periodic sine-cycloid method.

From this reason, a combined algorithm that produces the CoM trajectory in the forward direction using the periodic cycloid function and in the lateral direction using the optimal ZMP preview control algorithm was proposed by the author [41]. This method has two benefits; tuning CoM acceleration and keeping the flexibility of motion planning. Figure 4.3 and Figure 4.4 depict the resulting dynamic gait patterns.

4.2   Static-like Biped Motion Planning

Static walking is one strategy that people often employ to deal with uncertainties when walking. Compared with dynamic walking, it is slow but statically stable, which provides enough time to gauge the terrain and make gait adjustment. In humanoids, this strategy has been used in several applications. [20] introduced a static walking gait for the full-sized humanoid, THBIP-1. They designed feedback controllers and
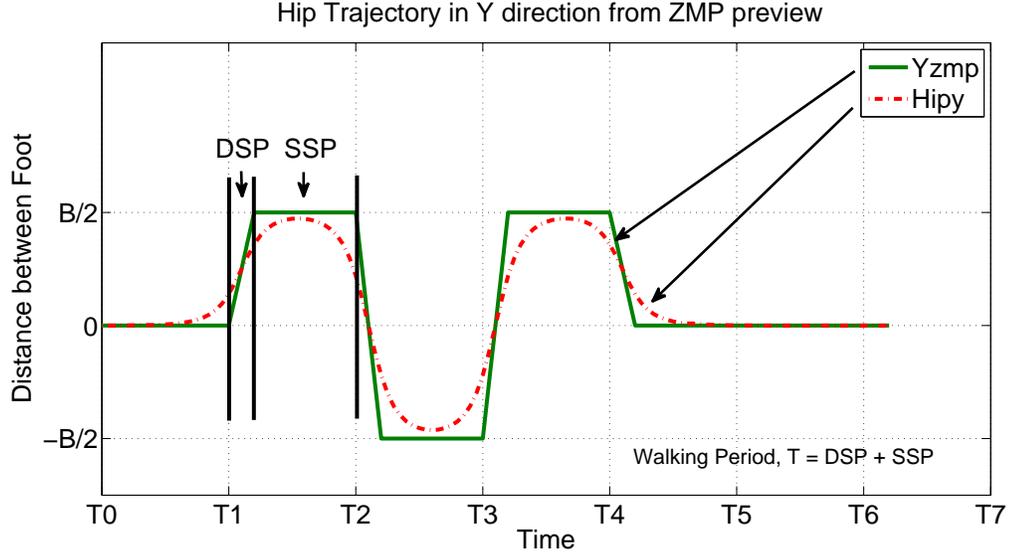
Figure 4.4: Dynamic walking gait for lateral (y) walking generated by optimal ZMP preview control.

validated its robustness against external disturbances through stair climbing experiments. [23] realized a static walking gait by applying resolved momentum control [48]. They planned a feasible foot trajectory for stepping over a given obstacle. The study was verified through experiments with a full-sized humanoid, HRP-2. For increased adaptability in maneuvering, the author plans a static-like gait patterns [44].

The ZMP equations derived from the 3D-LIPM, Eq (3.16) for x-direction and Eq (3.17) for y-direction, indicate that the gravitational force and inertial force are acting in the opposite way. To locate the ZMP in the center of support polygon which is typically set as origin, $(0, 0)$, such forces must be counterbalanced. In static walking, the gravitational term is dominant. Assuming that the inertial term can be ignored. This yields $x_{zmp} = x_{com}$ and $y_{zmp} = y_{com}$ in the horizontal plane. Thus, the CoM trajectory can be obtained from the desired ZMP trajectory.

The ZMP trajectory is a trace of points on the ground where the sum of the moments about the x- and y- axes are zero. The ZMP is typically assumed to be located
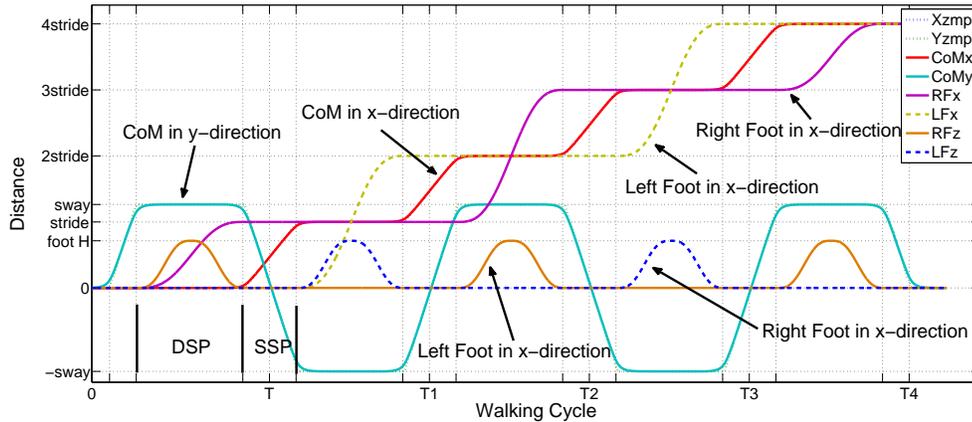
Figure 4.5: Static-like walking trajectories. The CoM and foot trajectories are indicated by arrows. The CoM trajectories are identical to the desired ZMP trajectories in x- and y-directions.

at the center of the support polygon. Knowing the feet positions and orientations, the desired ZMP trajectory can be manually planned or derived (see Section 3.1 and 3.3). Figure 4.5 depicts the CoM trajectories in the horizontal plane where T is a walking period that equals the sum of the DSP and SSP. To be clear, both right and left foot trajectories for x- and z-directions are also presented in Figure 4.5.

In Figure 4.5, the CoM trajectory is identical to the desired ZMP trajectory.

4.3   Phase Decomposition and Gait Planning for Uneven Terrains

Walking basically consists of a Single-Support Phase (SSP) and a Double-Support Phase (DSP). Horizontal walking can be designed based on the SSP and DSP described in the previous section. Likewise, the vertical CoM motion is fundamentally planned using the SSP and DSP but such phases should be subdivided when planning the vertical motion.

Different phases associated with climbing up and down stairs are notionally depicted in Figure 4.6. From this figure, one observes that a person's body moves up

Figure 4.6: Walking phase decomposition for the vertical CoM motion based on human obstacle climbing up/down motion

vertically between the DSP and SSP and moves down between the SSP and DSP. To capture this phenomena, the Upward Single-Support Phase (USSP) and Downward Single-Support Phase are defined. Additionally, the DSP is divided into two phases called Landing Double-Support Phase (LaDSP) and Lifting Double-Support Phase (LiDSP) for the control purposed explained.

Therefore, the reliable CoM motion in vertical direction can be designed based on 4 phases described in below.

- LiDSP - Lifting Double-Support Phase: This is the first phase in the walking cycle. The humanoid is supported by two feet. In this phase, the CoM moves to center of pressure of the support foot from the center of pressure between left and right foot in the DSP. Thus no vertical motion exists.

- USSP - Upward Single-Support Phase: In this phase, one foot is lifted up and starts to swing. If the humanoid elevated in the previous walking cycle, the

Figure 4.7: The vertical CoM trajectory (bold line) corresponding to the motion shown in Figure 4.8. $LFinit$ and $RFinit$ stand for the initial left and right foot position in the Y direction, respectively. Obs_H is the height of the obstacle and T denotes the walking cycle.

amount of elevation is compensated by the vertical CoM motion. Otherwise, no vertical motion exists.

- DSSP - Downward Single-Support Phase: The foot lifted in USSP is still swing. If the humanoid is required to climb down in the next foot step, the CoM moves vertically down in this phase. Otherwise, no vertical motion exists.

- LaDSP - Landing Double-Support Phase: The foot landed in DSSP and thus the humanoid is supported by two foots. In this phase, the CoM moves to the center of the support polygon formed by the left and right foot. Thus, there is no vertical CoM motion.

Based on the 4 phases and given the desired vertical CoM motion, a reliable CoM trajectory in the Z direction can be generated. To clarify the motion, let a humanoid take a step, place a foot on an obstacle, and step over the obstacle. Such motion is
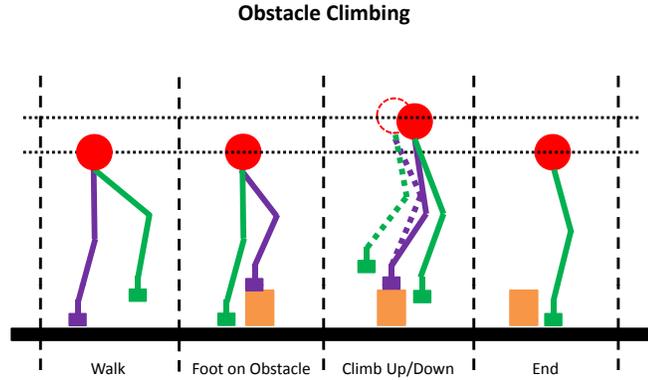
Figure 4.8: The obstacle climbing up/down motion corresponding to the vertical CoM trajectory in Figure 4.7.

depicted in Figure 4.8. According to the phase definition, the vertical CoM motion exists in the USSP and DSSP. With these ideas, the CoM trajectory is designed and shown in Figure 4.7 corresponding to Figure 4.8. In this motion, a third-order interpolation method is applied to smoothly generate the vertical CoM motion.

In the designed three dimensional CoM motion, the humanoid quickly transitions from walking to climbing up/down and vise-versa. Based on the CoM trajectory and climbing height, the foot trajectory can be generated using the cycloid method described in Section 3.6.

The horizontal foot motion can be designed using Eq (3.38) and Eq (3.39). For the foot motion in the Z direction, a new parameter, Obs_H, is added to Eq (3.40) and Eq (3.41) and H becomes an offset. Obs_H denotes the obstacle height. In the design of the foot motion, timing is the most important consideration. Since the CoM and foot motion are produced independently based on a given ZMP trajectory, both horizontal motion are synchronized. But the desired ZMP trajectory is a two dimensional trace
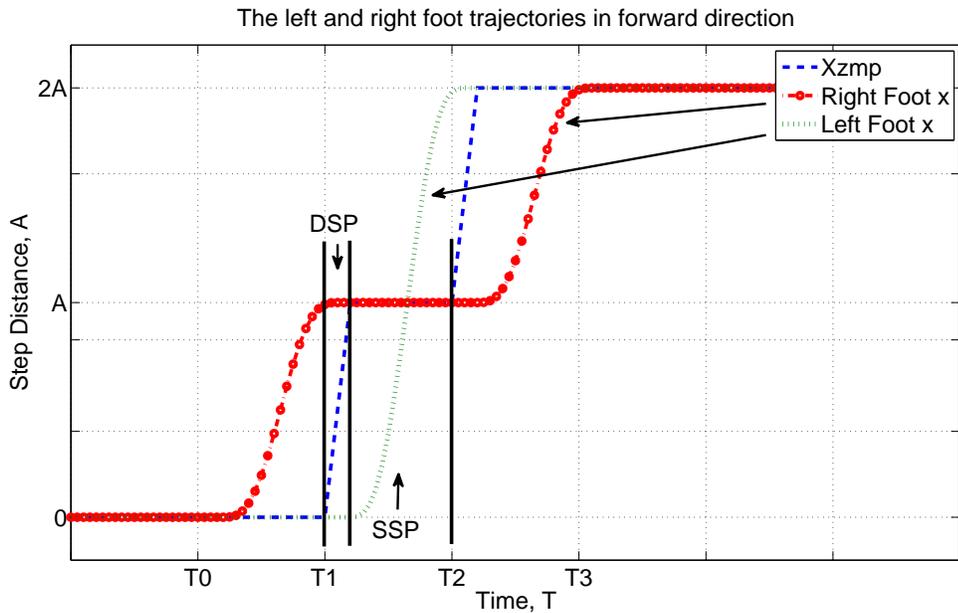
Figure 4.9: Left (light dotted line) and right (circle-dotted line) foot trajectories in the forward direction with a given ZMP trajectory in X direction (bold dotted line), walking period ($SSP + DSP$), and step distance (A)
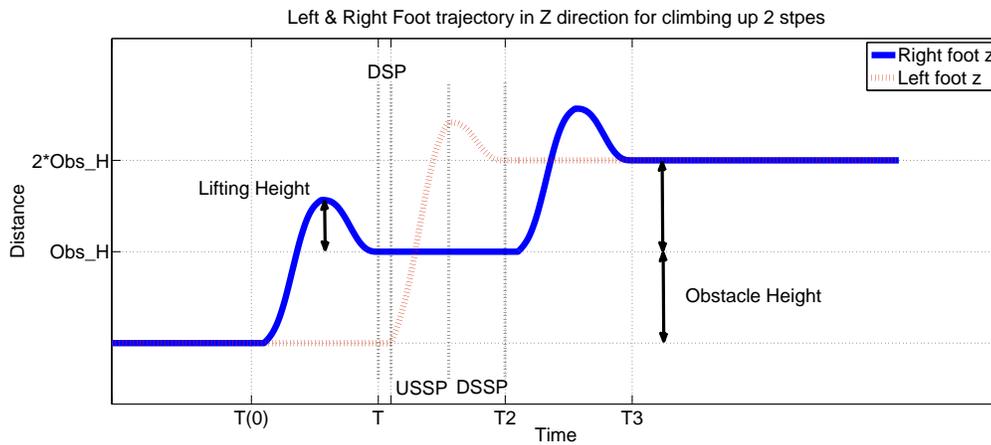
Figure 4.10: Left (light dotted line) and right (bold line) foot trajectories in the vertical direction, Z, while obstacle climbing. Obs_H and T denote the obstacle height and walking cycle including DSP, $USSP$, and $DSSP$, respectively.
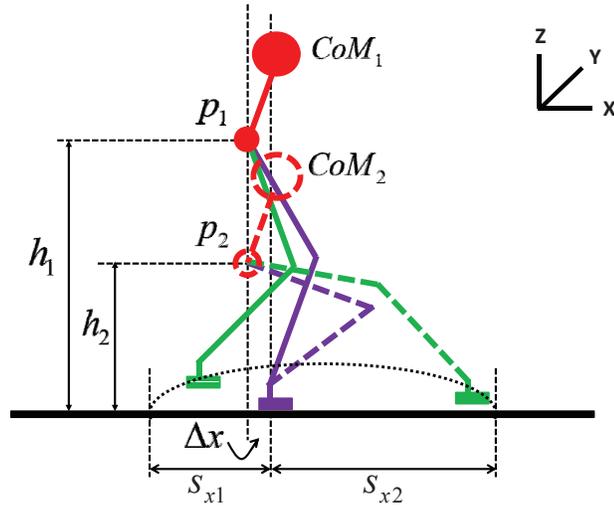
Figure 4.11: Pelvis height adjustment: The pelvis height is adjusted in the Single-Support Phase based on an input stride in order to avoid a singularity in the IK problem and improve the power efficiency in the static walking strategy.

and thus both vertical timings cannot be matched. Using the decomposed phases, the vertical CoM and foot motion are also synchronized.

Figure 4.9 shows the left and right foot trajectories for two steps in the forward direction. A represents the forward step distance and T is the walking cycle. For the foot motion in Z direction, Figure 4.10 depicts the left and right foot motion in that direction for climbing up.

4.4 Pelvis Height Adjustment

The stable pelvis, calculated from the CoM position, and foot trajectories are planned in Cartesian space. The IK solver converts such trajectories to joint configurations to drive actuators. Humanoids often crouch when walking in order to increase reachability without encountering a singularity in the IK equations. Stepping further requires a lower pelvis height to avoid unsolvable IK problems. Crouching in walking,

however, consumes more power. With simple trigonometric functions, the humanoid adjusts the pelvis height according to the step distance.

Figure. 4.11 depicts a humanoid in the forward (x-) direction in the Single-Support Phase of static walking. p, h, and s are the pelvis, pelvis height, and stride, respectively. Subscripts 1 and 2 denote the first half and last half of the SSP periods, respectively. Δx is the difference in distance between the CoM and pelvis positions in the forward direction. Δx can be easily calculated with the given CoM trajectory and torso's pitch angle.

Let l_{leg} be the length of a leg. Assume that the humanoid initially crouches by h_{crouch} in the vertical direction. h_{crouch} is a negative constant manually determined and used in walking. With an input stride, s_{x2}, kinematic length of a leg, l_{leg}, and initial displacement of the pelvis, h_{crouch}, the pelvis height for the next foot step can be obtained in (4.1).

$$h_2 = \sqrt{(\Delta x + l_{leg})^2 - s_{x2}{}^2} + h_{crouch} \tag{4.1}$$

In Fig. 4.11, the robots outlined in bold and dotted lines have h_1 and h_2 derived from the previous (s_{x1}) and next foot (s_{x2}) stride, respectively. The pelvis motion in the vertical direction is done in the first half of the USSP period. The motion is planned using a third-order interpolation method. The humanoid autonomously adjusts the pelvis height corresponding to the given stride.

4.5   Foot Placement for Path Following

In the previous sections, the horizontal biped locomotion is realized based on the desired ZMP trajectory and the vertical motion is implemented according to the phase defined. But for the practical application, the humanoid must generate the reliable biped locomotion in real time with the visual sensory data. It would be the ideal if
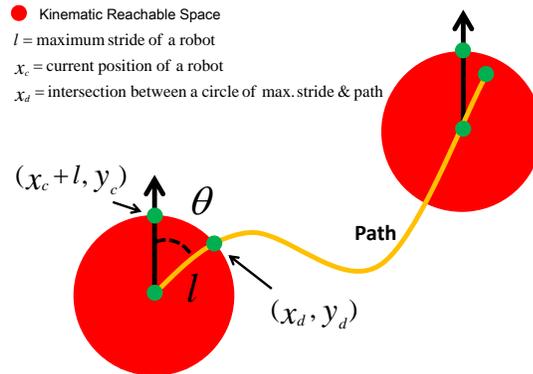
Figure 4.12: A case for a humanoid to follow a desired path defined by a path planning algorithm.

the measured data gives the X, Y, and Z positions and the rotation, θ, for the next foot step. In this case, the reliable walking trajectory can be simply obtained based on author's approach explained in the previous sections.

However, in some cases the data might be a target position in terms of the global frame or a desired path to follow defined by one of the path planning algorithms such as A-star, Rapid Random Tree (RRT) [64], and Potential Field [54]. In this case, the humanoid must determine the proper stride to proceed and the angle to turn, accordingly.

Let us assume that an arbitrary path on the ground is given by a path planning algorithm and the humanoid is currently heading in the forward direction pointed by an arrow in Figure 4.12. In Figure 4.12, l denotes the maximum stride of the humanoid. It can be kinematically limited by the length of legs. The coordinates (x_c, y_c) and (x_d, y_d) are the current position of the humanoid and the position of intersection between the path and kinematic reachable space (red circle) defined by l, repetively. Then the target angle, θ, can be obtained from Eq (4.2) and Eq (4.3).

$$l^2 = (x_d - x_c)^2 + (y_d - y_c)^2 \tag{4.2}$$

$$\frac{(x_c + l, y_c) \times (x_c, y_c)}{l^2} = \sin(\theta) \tag{4.3}$$

But here are some constraints.

- If θ is too large for the humanoid to rotate at once, then the humanoid turns in place until its orientation coincides with θ.

- If the intersection point, (x_d, y_d), does not exist, then it is assumed that the humanoid reaches the target. Otherwise, reduce l until the intersection point is found.

Note that l does not need to be the maximum stride but any reachable stride.

4.6   Obstacle Avoidance

Obstacle avoidance challenge is a case study to verify and validate the performance of real-time gait planner. The gait planner produces the pelvis and foot trajectories based on visual feedback. A block diagram of the overall control is depicted in Figure 4.13.

The dot box represents the real-time gait planner where the periodic sine-cycloid and optimal ZMP preview control generate the stable CoM trajectories in the x- and y-direction, respectively. At every single step, the robot processes the visual data for the next foot step which returns the distance to travel and angle to turn. The ZMP-Foot Generator generates the reference ZMP trajectory and foot motion related to the visual processing data. The reference ZMP trajectory is used as the input to the real-time planner. The resulting CoM and foot motion planned in Cartesian space
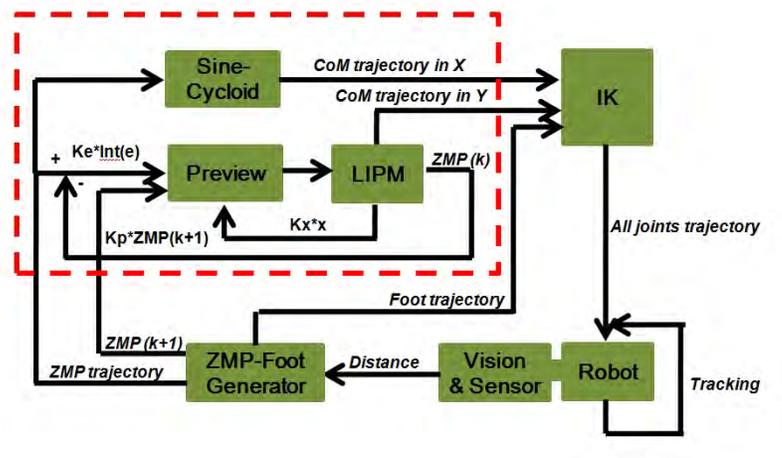
Figure 4.13: Control block diagram of real-time walking planner for obstacle avoidance.

are converted to joint angle trajectories via vector IK. In this case, the upper-body orientation is set as upright to the ground.

Figure 4.14 represents two scenarios that the humanoid faces in the obstacle field. Figure 4.14(a) is a case where no obstacle is detected in the path and Figure 4.14(b) is a case when an obstacle is on the path. In the vision algorithm shown in Figure 4.15, there are nine points in an image; three points for right, center, and left sides. If the sum of RGB values of all points equals to zero, it turns out no obstacle on the path. Then, the humanoid go straight. Likewise, the robot side-steps or turns to right when the sum of RGB values in the left column is greater than the right column. Otherwise, the robot side-steps or turns to left.

A miniature humanoid, Mini-Hubo, is used for the experiments. The SSP and DSP are set as 0.2 and 0.8 seconds, respectively. Figure 4.16 depicts two gait patterns used for the robot to walk forward and side-step, respectively. Figure 4.16(a) and (b) are the foot motion, Figure 4.16(c) and (d) represent the pelvis motion in forward direction, and Figure 4.16(e) and (f) are the pelvis motion in lateral direction

(a) Case 1: Walk forward   (b) Case 2: Side step or turn
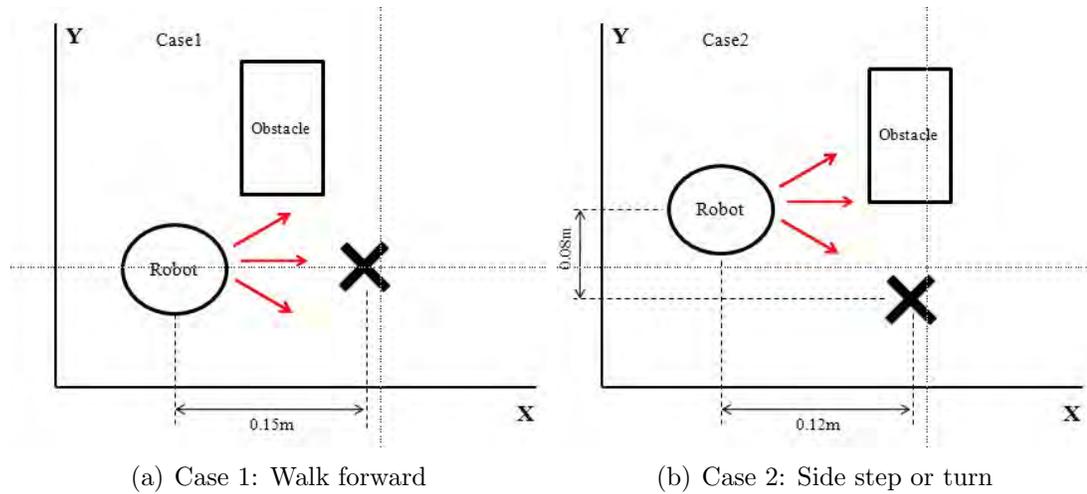
Figure 4.14: Simple visual commands for the robot steering.
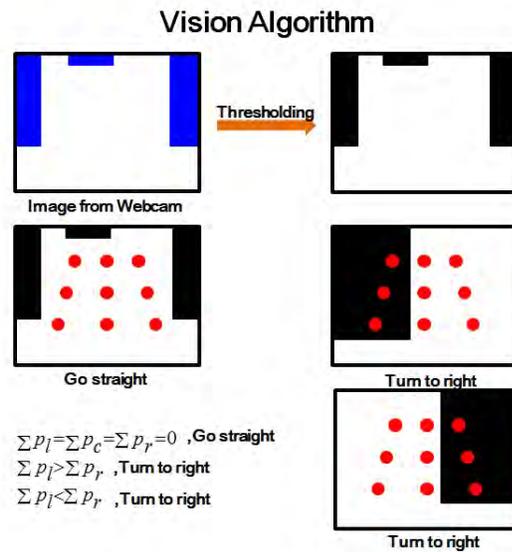
Figure 4.15: Image processing for path following.
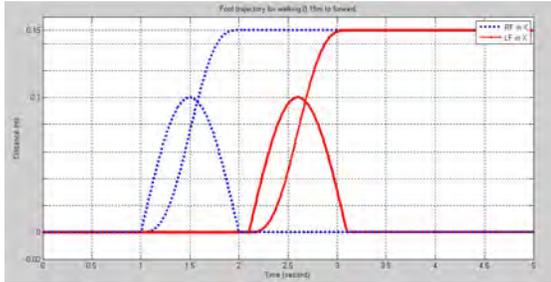
(a) Right and left foot trajectories for walking forward.

(b) Right and left foot trajectories for side-stepping. The foot moves in both x- and y-directions.
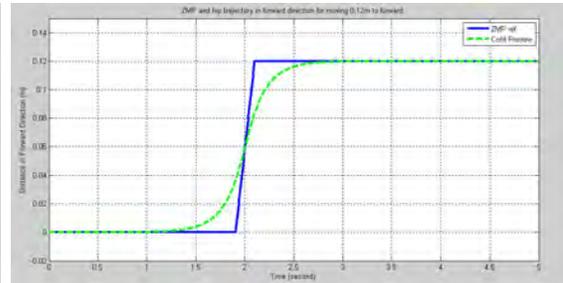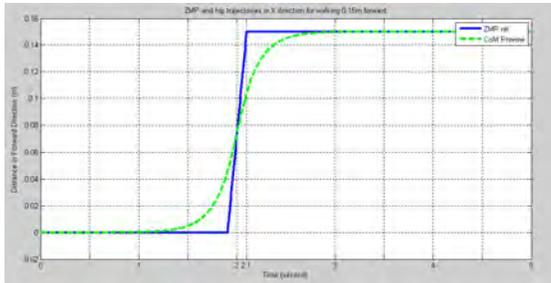
(c) Pelvis in x-direction for walking forward.

(d) Pelvis in x-direction for side-stepping.
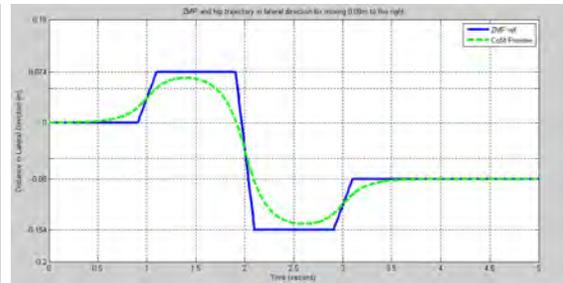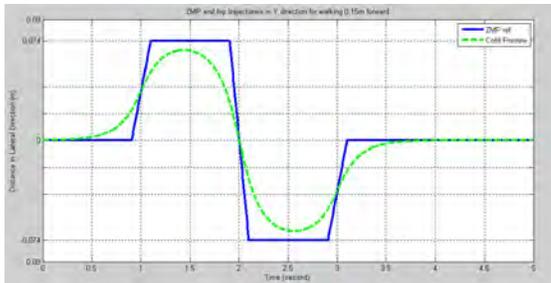
(e) Pelvis in y-direction for walking forward.

(f) Pelvis in y-direction for side-stepping.

Figure 4.16: Gait patterns for case 1: walking forward (a), (c), and (e). Case 2: side-stepping (b), (d), and (f).
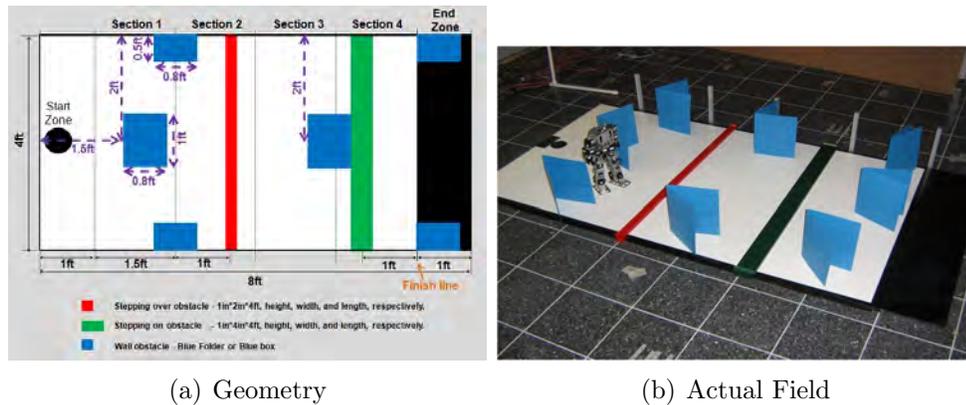
(a) Geometry   (b) Actual Field

Figure 4.17: Obstacle Field

according to a given ZMP trajectory.

The obstacle field for experiments is shown in Figure 4.17 and Figure 4.18 is a snapshot of mini-Hubo walking in the obstacle field. In this experiment, the stepping-over obstacles were removed. Mini-Hubo successfully avoided all obstacles and reached the goal location.

4.7 Stair Climbing

To verify and validate author's proposed method for locomotion on uneven terrains, the stair climbing is selected as a case study. Stair climbing requires a relatively large foot step to avoid the collision with the stairs. To minimize the linear momentum in walking, the combined method described in Section 4.1 is used to generate the CoM trajectory. Foot motion is planned with cycloid method demonstrated in Section 3.6. Figure 4.19 shows all trajectories for stair climbing.

The stair climbing was rapid prototyped in virtual-Hubo, tuned and evaluated with mini-Hubo, and verified and validated on online-Hubo. The control frequency of each platform was 50Hz, 50Hz, and 200Hz, respectively. To overall idea verification, physical stairs were setup based on the ratio of dimension between mini-Hubo and
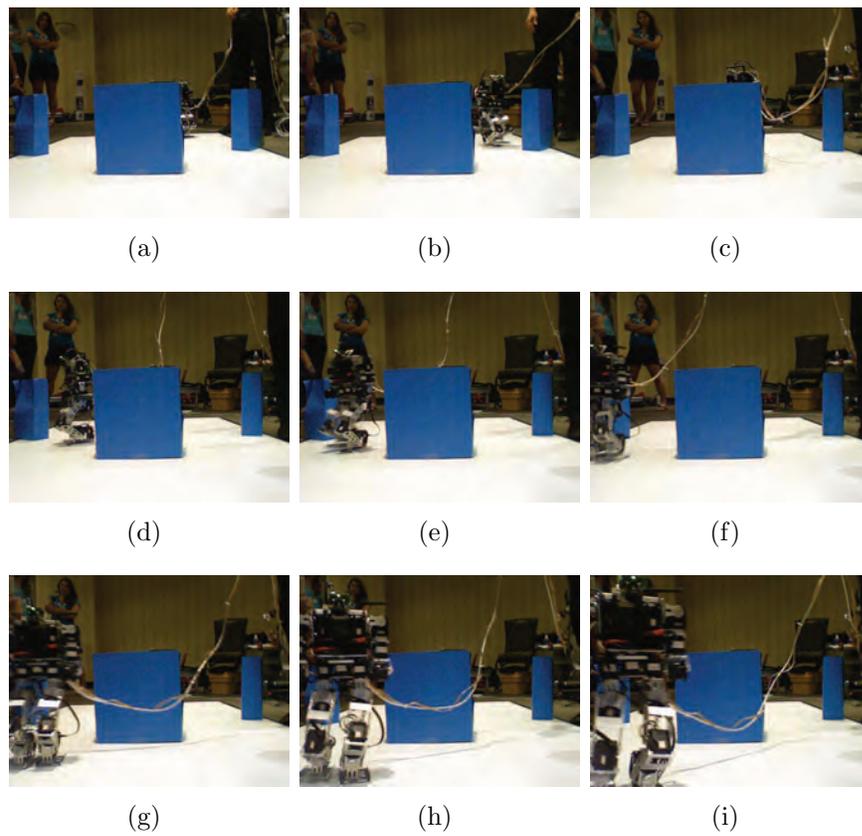
Figure 4.18: Mini-Hubo obstacle avoidance.
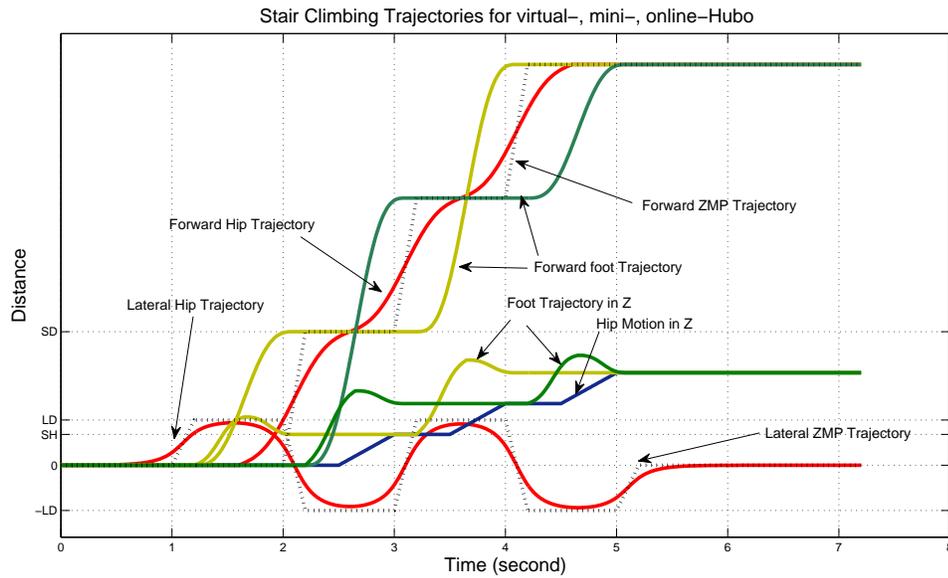
Figure 4.19: Stair climbing trajectories for CoM and foot tested on 3-Tier infrastructure

online-Hubo. Figure 4.20 is a snapshot of virtual-Hubo climbing up stairs in simulation. Figure 4.21 and Figure 4.22 are a snapshot of experiments with mini-Hubo and online-Hubo, respectively.

4.8   Whole-Body Locomotion on Unstructured Terrains

Humanoid's capability to locomote unstructured terrains toward the goal of working in outdoor environments is tested and evaluated. Assume that the geometric information of terrains like incline, height, and location is measurable by visual sensors such as laser scanners. Figure 4.23 is a snapshot of online-Hubo walking on unstructured terrains. The gait planner for the pelvis motion in the x- and y-directions, pelvis height adjustment, cycloid foot motion, and upper-body compensator (see Section 5.3) were fully integrated for this experiment. The stride for right foot was $[260 \quad 300]mm$ and for left foot was $[220 \quad 0]mm$. Corresponding height for right foot
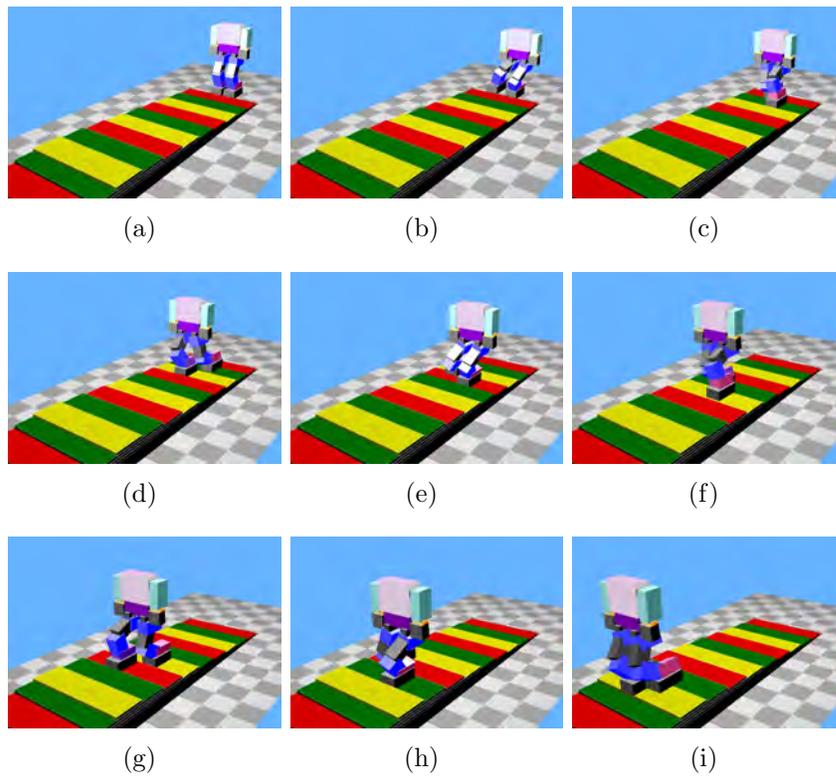
Figure 4.20: Vitual-Hubo walking on stairs.
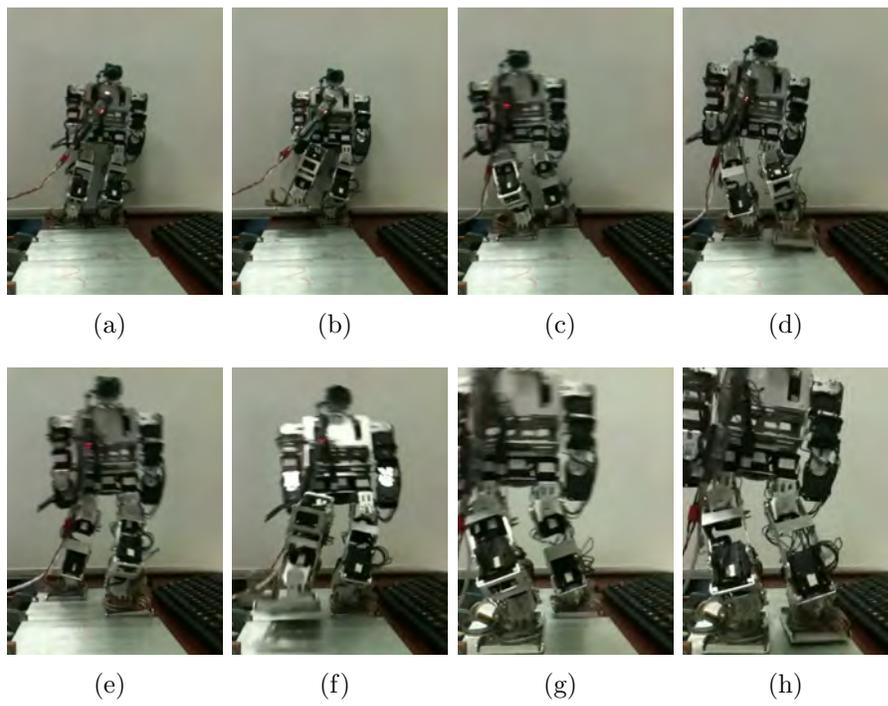
Figure 4.21: Mini-Hubo climbing stairs.
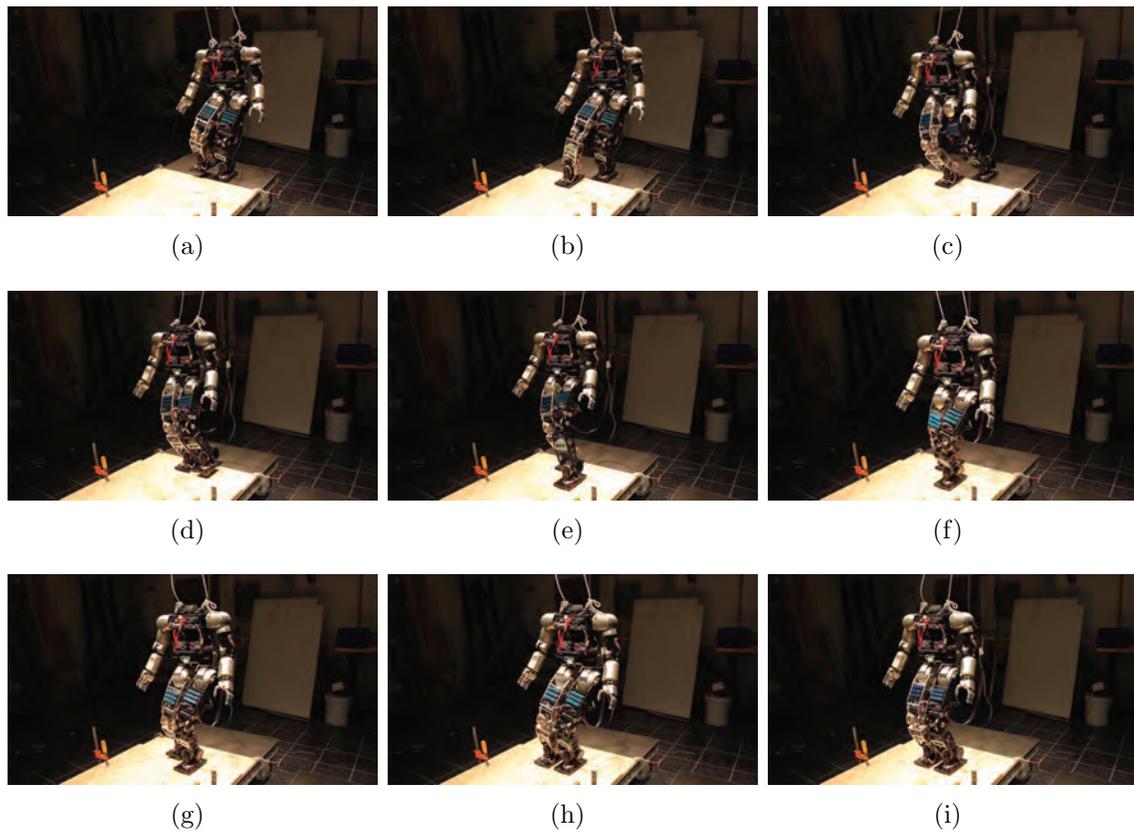
Figure 4.22: Online-Hubo walking on stairs.

was $[-40 \quad -120]mm$ and for left foot was $[40 \quad 0]$. The SSP period was 6 seconds and DSP time was set as 3 seconds. The foot height was $50mm$. Figure 4.24(a) shows the ZMP and CoM trajectories in the x- and y-directions, generated from the combined approach explained in Section 4.1. Figure 4.24(b) depicts the foot motion corresponding to the given foot parameters.

Figure 4.25(a) shows the crouching distance in the vertical direction. Such pelvis height was adjusted based on a given stride. h_{crouch} was $-50mm$ in Eq 4.1. Figure 4.25(b) depicts the upper-body orientation. A feedforward controller that compensates the torque exerted on the support foot by the upper-body motion in the opposite direction for counterbalancing.

Figure 4.23: Online-Hubo walking on uneven terrains.
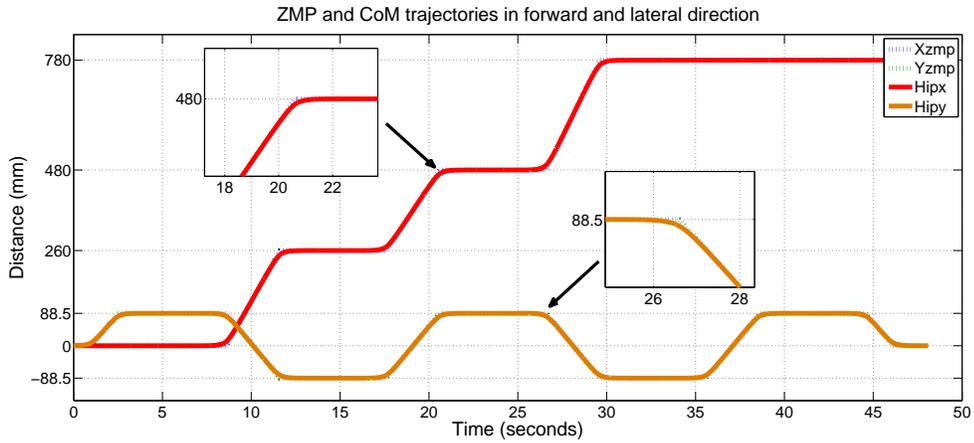
(a) ZMP and CoM trajectories in x- and y-directions planned with the given information of strides
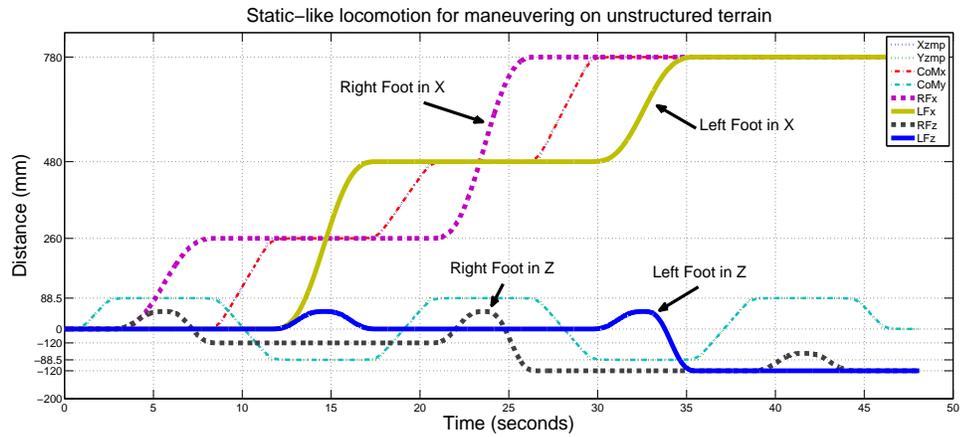
(b) Foot motion in x- and z-directions corresponding to Figure 4.24(a)

Figure 4.24: Gait patterns for walking on unstructured terrains.
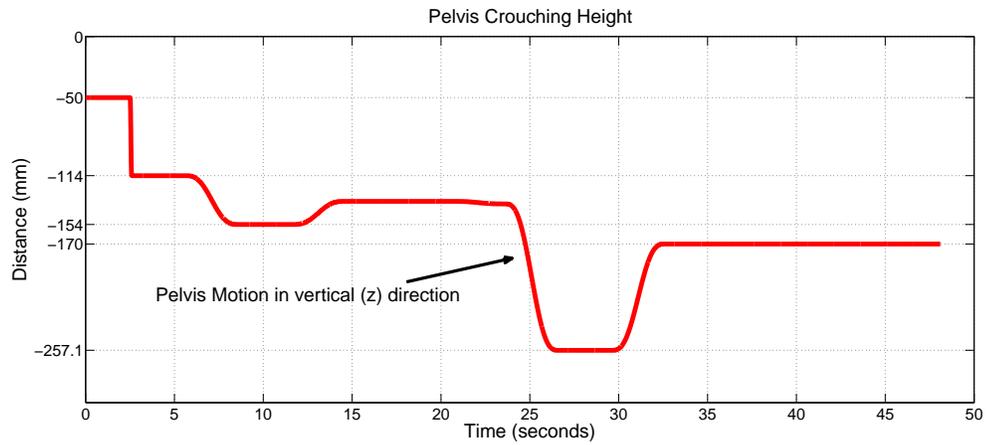
(a) Pelvis height adjustment
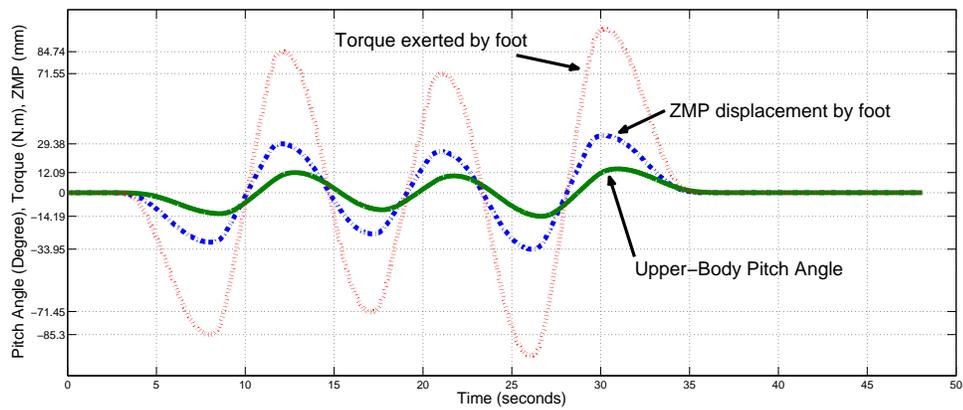
(b) Upper-body orientation (pitch angle) in forward locomotion.

Figure 4.25: Two feedforward controls for locomotion on unstructured terrains.

5. REAL-TIME FEEDBACK MOTION CONTROLLER

The open-loop locomotion and manipulation planned for humanoids often results in kinematic and dynamic errors because of discrepancies in modeling, mechanical uncertainties, and sensory noise. In addition, environmental disturbances like surface roughness and wind force often significantly influence the dynamics of humanoids. Such internal and external disturbances are often arbitrary and but are critical to maintain balance. Humanoids have a relatively small stable area and lack compliance. The forces from such disturbances acting on the body directly transfer to the moment on the support foot. For reliable, stable, and adaptable motion, real-time controllers that compensate such disturbances are required for the stable motion planning.

In this chapter, these real-time controllers are presented. These controllers allow a humanoid to balance against reactionary forces in manipulation, minimize the dynamic influences from internal and external disturbances, and provide active compliance to mitigate the impulsive contact forces. The performance of these controllers are tested and evaluated experimentally using a full-sized humanoid.

5.1 Linear Quadratic Integrator

For stability, the ZMP must stay within the support polygon. However, humanoids lack compliance. A small disturbance acting on the body produces a large moment about the foot. In static gait planning, the CoM motion is the only consideration. But this does not fully represent the actual dynamics of a humanoid. To compensate for perturbations of the ZMP by internal and external disturbances, a Linear Quadratic Integrator (LQI) is implemented.

The LQI technique is a classic control law that has been used for reference tracking
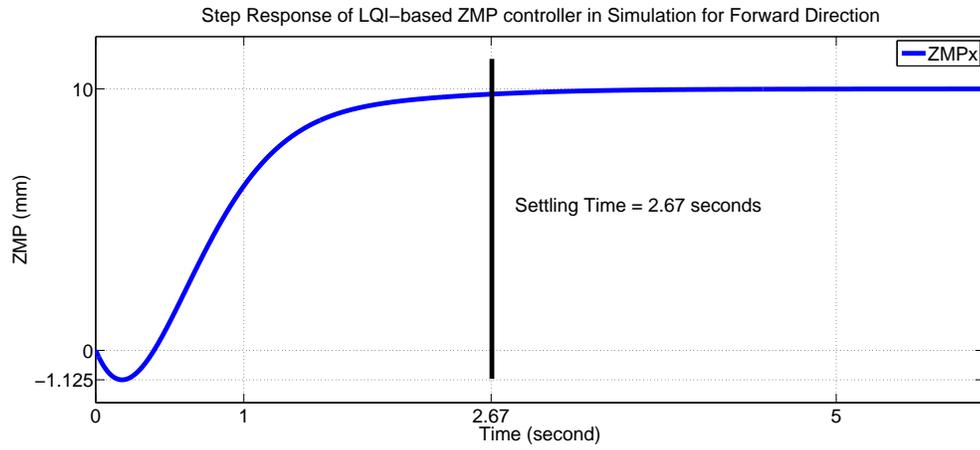
Figure 5.1: Simulated step response of DRC-Hubo with LQI controller in forward (x-) direction.
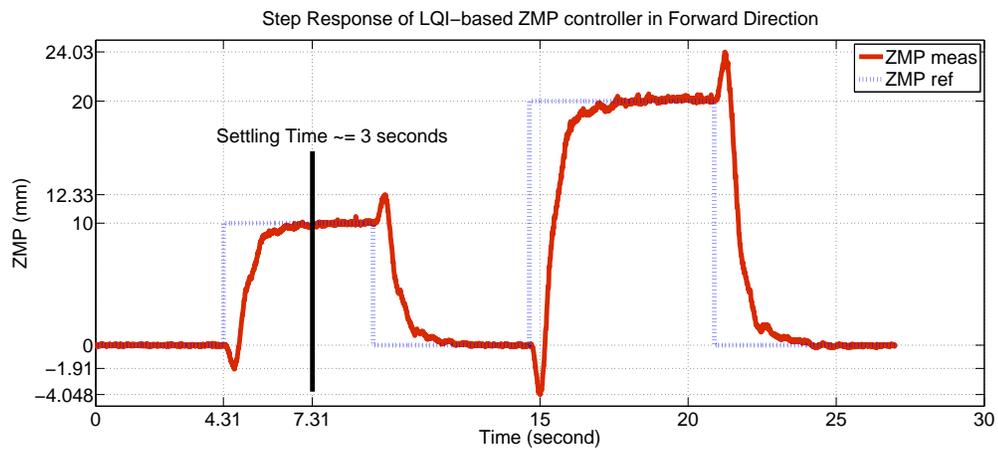
Figure 5.2: Actual step response of DRC-Hubo with LQI controller in forward (x-) direction where the dotted and bold lines represent the reference and measured ZMP, respectively.

systems. One of the outstanding applications of LQI in humanoids is demonstrated in [47]. This study utilized the previewable LQI technique [53] based on LIPM for dynamic gait planning. According to [47], the discrete state-space form of LIPM with the ZMP output and performance index that needs to be minimized are represented in Eq (3.22) and Eq 3.23 in Section 3.4, respectively.

Q_x is a 3×3 symmetric semi-definite matrix shown in Eq (5.1) where W is the weight value.

$$
Q_x = \begin{bmatrix} W_1 & 0 & 0 \\ 0 & W_2 & 0 \\ 0 & 0 & W_3 \end{bmatrix}
\tag{5.1}
$$

By solving the Discrete Algebraic Riccati Equation (DARE), the optimal control input of the LQI, u, is obtained in Eq (5.2).

$$
u(k) = K_p(-G_i \sum_{i=0}^{k} e(k) - G_x X(k))
\tag{5.2}
$$

where K_p is a proportional gain for tuning and G_i and G_x are the optimal gains from DARE.

The LQI-based ZMP controller in forward (x-) direction is designed with the following parameters; $z_c = 0.66$, $\begin{bmatrix} Q_e & R \end{bmatrix} = \begin{bmatrix} 1 & 1.0 \times 10^{-6} \end{bmatrix}$ and $\begin{bmatrix} W_1 & W_2 & W_3 \end{bmatrix} = \begin{bmatrix} 10000 & 10 & 100 \end{bmatrix}$. The ZMP controller for lateral (y-) direction has the same parameters as above except that W_1, W_2, and W_3 are 100, 10, and 100, respectively.

The desired and actual step responses for 10mm ZMP step input in x-direction are plotted in Figure 5.1 and 5.2, respectively. The actual settling time and undershoot are approximately 3 seconds and 20% while the desired values are 2.67 seconds and 11%, respectively.

Figure 5.3: A posture at the beginning of SSP. The weight of the leg lifted displaces the ZMP in the support foot by a certain amount proportional to mass of the leg (m_r) and stride (x_r).

5.2   ZMP Displacement Control via Reference ZMP Adjustment

The leg is typically a heavy part in humanoids. It contains powerful actuators and strong frames to support the entire body. While the LIPM is mathematically simple and easily implemented, it does not account for the dynamics of the legs. In the static-like locomotion, the gravitational force on the legs becomes more dominant in balancing as the foot steps further from the CoM. The sway of the leg displaces the ZMP location. The LQI-based ZMP controller in Section. 5.1 is designed to handle such ZMP variations. However, it is a low frequency controller that cannot quickly respond to a sudden ZMP displacement.

When the rear foot is lifted up at the beginning of SSP, the moment induced by the weight of the swinging leg rapidly displaces the ZMP. Figure 5.3 depicts such phenomena at the moment of transition from DSP to SSP where Δzmp represents the ZMP displacement. To estimate such ZMP displacements, the author represents

a humanoid as a model consisting of three point masses, m_u, m_r, and m_f shown in Figure 5.3. The subscripts u, r, and f denote the upper-body, rear, and front, respectively. From this model, the torque, τ_r, acting on the supporting foot, which is generated by the swinging leg in x-direction, can be obtained from Eq (5.3).

$$\tau_r = m_r((\frac{l_r}{2})^2 + \frac{l_r l_p}{2})\ddot{\theta}_r + \frac{m_r g l_r}{2}\sin(\theta_r) \tag{5.3}$$

where l_r and l_p are the length of rear leg and pelvis height. θ_r denotes the angle between the vector of l_r and the vertical axis.

At the instant of the foot being lifted, the moment generated by the gravitational term is dominant. Let $l_r\theta_r$ be x_r representing the stride in the x-direction. As a result, the ZMP displacement in the x-direction can be calculated from Eq (5.4) based on the ZMP definition, $\frac{-\tau}{f_r}$ where f_r is the reaction force.

$$\Delta ZMP = -0.5x_r\frac{m_r}{m_t} \tag{5.4}$$

where m_t denotes the total mass that equates to $m_u + m_r + m_f$. In the same manner, the ZMP displacement in y-direction can be obtained.

According to Eq (5.4), the ZMP displacement is proportional to the foot stride and mass of the leg. This signifies that a longer step size results in a significant displacement of the ZMP; possibly moving out of the support polygon. For stability, it is required that such a ZMP displacement is compensated for. A motion that counterbalances the torque in Eq (5.3) can handle such ZMP displacement [25][42]. In locomotion, the LQI-based ZMP controller tracks the reference ZMP trajectory in real time. The same ZMP displacement magnitude is added to the reference ZMP trajectory in the DSP. Then, the torque produced by the CoM displacement resulting from the LQI controller keeps the ZMP within the support polygon from the ZMP

(a)

(b)
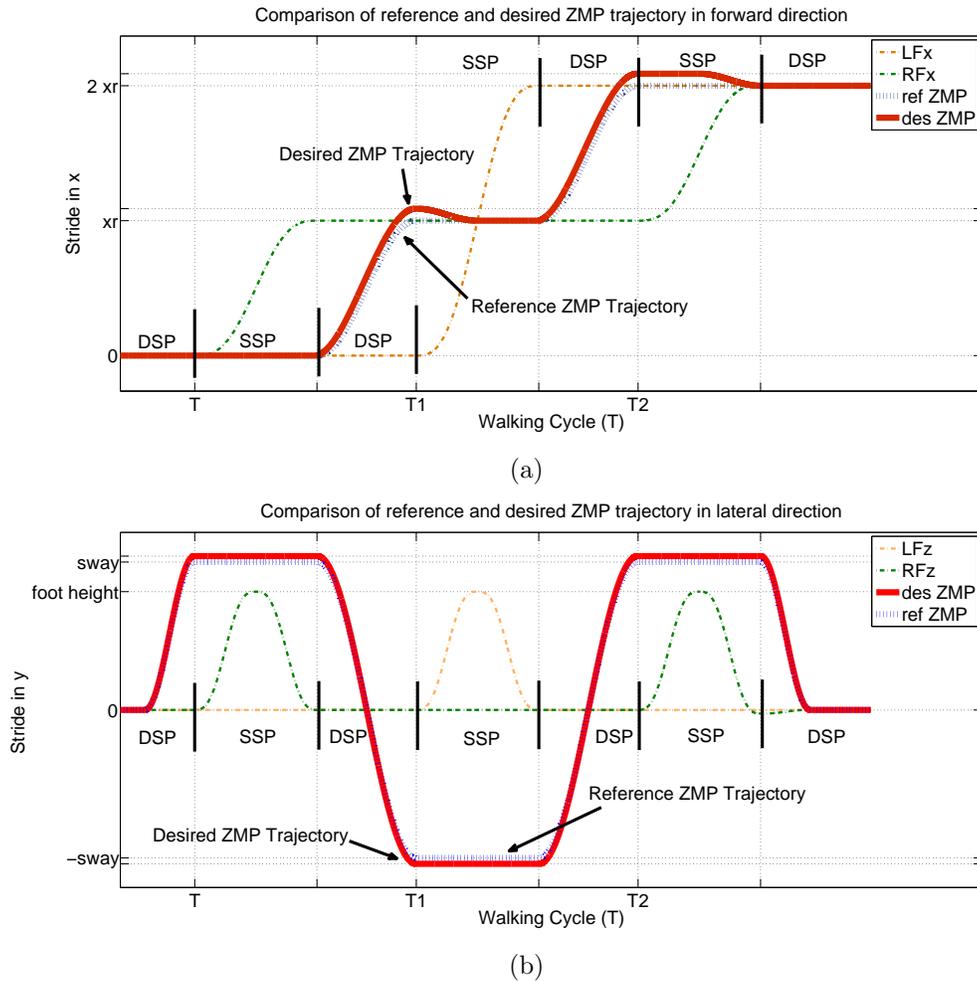
Figure 5.4: Desired and reference ZMP trajectories for two steps in the forward direction, indicated by arrows. The light dotted lines in (a) and (b) represent the right and left foot trajectories for the x-direction and z-direction, respectively. T denotes the walking period. Those trajectories were actually obtained from DRC-Hubo walking with two steps (x_r=390mm) in the forward direction on an even surface.

displacement.

Figure 5.4 (a) and (b) show a visual example of the desired ZMP trajectory in the x- and y-direction phase by phase, respectively. The desired ZMP trajectory for the DSP is obtained from Eq (5.5).

$$ZMP_{des} = ZMP_{ref} - \Delta ZMP \sin(\frac{\pi f}{2}t) \tag{5.5}$$

$$\text{where } 0 \leq t \leq DSP$$

After the rear foot is lifted, the ZMP displacement decreases as the foot swings. The desired ZMP trajectory for the SSP is defined in Eq (5.6).

$$ZMP_{des} = \{ \begin{matrix} ZMP_{ref} + \Delta ZMP & \text{if } \Delta ZMP < 0 \\ ZMP_{ref} & \text{otherwise,} \end{matrix} \tag{5.6}$$

Note that this control strategy is not only for locomotion but can be applied to manipulation of a heavy object. In manipulation, the reactionary forces in interaction with an object often yield such ZMP displacements [25]. Such reactionary forces are typically measurable by force sensors located in the wrist joints. The torque exerted on the support foot in Eq 5.3 can be obtained with the measured forces and kinematic positions of hands. The ZMP displacement from the applied torque can be compensated by adjusting the reference ZMP in the same manner as previously explained in this section.

(a) Upper-body upright posture   (b) Upper-body compensator
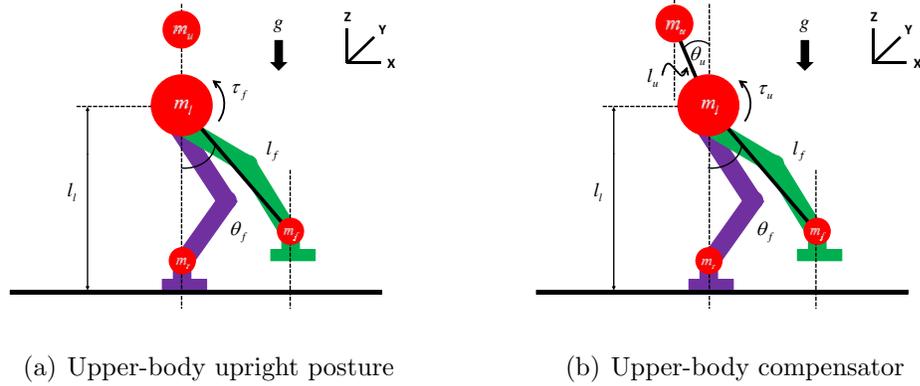
Figure 5.5: Dynamic model of humanoids used for upper-body compensator. The model is based on the Double Inverted Pendulum (DIP).

5.3   Upper-Body ZMP Compensator

Another control strategy can be used to compensate the ZMP displacement from known/measurable disturbances. It controls the ZMP location through the upper-body motion. Like a human, the physical capability of humanoids in a given foot configuration can be maximized through the upper-body posture (see Section 6.2). The inertial and gravitational forces of the upper-body are transferred to the moment on the support foot. This strategy allows humanoids to control the stability without the adjustment of ZMP reference.

Figure 5.5 shows the dynamic model used to design the upper body controller. Assume that the upper-body motion and foot motion are decoupled. The torque, τ_f, acting on the pelvis link exerted by the front foot (swing foot in Figure 5.5(a)) is linearized as

$$\tau_f = m_f(l_f^2 + l_l l_f)\ddot{\theta}_f + m_f g l_f \theta_f \tag{5.7}$$

where subscript f and l denote front foot and lower-body. m and l denote the

mass and length, respectively. θ represents the angle between the foot and vertical axis.

Let $l_f \theta_f$ be x_f. Then, Eq (5.7) becomes

$$\tau_f = m_f(l_f + l_l)\ddot{x}_f + m_f g x_f \tag{5.8}$$

where x_f is the relative position of the front foot in the x-direction from the CoM position in the same direction.

The torque acting on the pelvis link by the support foot, τ_r where subscript r denotes the rear foot, can be calculated in the same manner. Once τ_f and τ_r are obtained from the model, the ZMP displacement by both feet can be obtained using Eq (3.7).

$$\Delta ZMP_x = \frac{\tau_f + \tau_r}{m_t g} \tag{5.9}$$

where m_t is the total mass of the humanoid.

The control objective is to design an upper-body motion that compensates the torque acting on the pelvis link by τ_f and τ_r. From the model in Figure 5.5(b), the upper-body torque acting on the pelvis link, τ_u, can be defined as,

$$\tau_u = -m_u l_u g \theta_u + (m_u l_u + m_u l_l l_u)\ddot{\theta}_u \tag{5.10}$$

where the subscript u stands for the upper-body. θ_u denotes the pitch angle of the upper body in this case. The model in lateral direction can be obtained in the same manner.

Using the upper-body dynamics in Eq 5.10, the ZMP tracking control for Eq 5.9 can be designed via control laws.
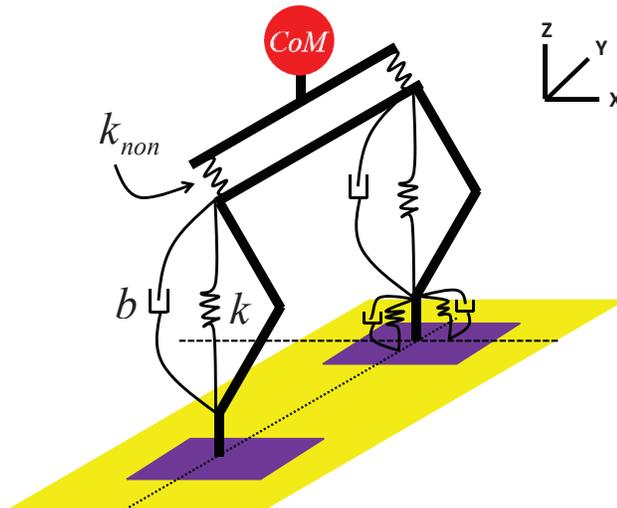
Figure 5.6: Virtual suspension for compliance control.

5.4   Lower Body Compliance Control

The uncertainties from sensory measurements and rigidity of the mechanical struc-
ture often hinder planning walking trajectories from perfectly matching the ground
topography. Because of the lack of compliance in humanoids, such kinematic errors
generate a large reactionary force that directly yields moments about the foot. In
locomotion, both feet are fully constrained to the ground in the DSP while the desired
motion transits the CoM from one foot to the other. Indeterminate internal mechan-
ical stiffness in the legs results in kinematic errors in this phase. Virtual compliance
is designed to not only absorb the impact force in the foot landing but also minimize
such phenomena.

The landing impact is a well known phenomenon that has been intensively re-
searched. Such works can be simply categorized into two approaches: passive and
active compliance. The passive compliance is a mechanical design like an artificial
foot [28][69] and low-stiffness joints [107]. It generally consists of physical springs,

dampers, and switches. Such components absorb the energy in landing and detect foot contact. The performance is robust but fully limited to its design. Conversely, active compliance virtually realizes the spring-damper system. Such virtual compliance is typically realized based on mathematic landing models and feedback control laws [29]. Its performance is adaptive to the surface condition but the gain tuning is extremely tedious and difficult.

The landing compliance consists of force and moment compliance controllers. The force compliance control minimizes the impact force while the moment compliance control properly maintains the foot contact with the ground. The virtual mass-damper-spring system is placed in the leg as shown in Figure. 5.6.

The governed equation for the virtual suspension is,

$$m\ddot{x} + b\dot{x} + kx = F_{meas} - F_{ref} \tag{5.11}$$

where m, b and k are the mass, damping, and spring coefficients, accordingly. F_{meas} and F_{ref} denote the force measured by the Force-Torque sensor (F/T sensor) and force profile. X is the displacement of the foot in the vertical direction. For moment compliance control, x becomes θ for the ankle pitch and roll joint.

The CoM is located in the support (rear) foot at the swinging (front) foot landing in static walking. Thus, the force profile of the rear and front foot can be simply defined as $F_{rf} = W_t$ and $F_{ff} = 0$, respectively, where the subscripts rf and ff denote the rear and front foot, and W_t is the total weight of the robot.

The compliance control block diagram is shown in Figure. 5.7. In landing earlier than expected, $F_{meas} >$ threshold, the leg is bending to absorb the impact force. Otherwise, it is assumed that the foot does not touch the ground if $F_{meas} <$ threshold. In the case of landing later than expected, the arbitrary negative force is applied to the controller in order to stretch the leg until it makes contact with the ground. The

$$x_z, \theta_{pitch,roll} \qquad\qquad\qquad F, M$$

$$m\ddot{x} + b\dot{x} + kx$$

Robot   F/T sensor

$FootTra_z,$

$AnkleTra_{pitch,roll}$ $\qquad\qquad F_{ref}, M_{ref}$
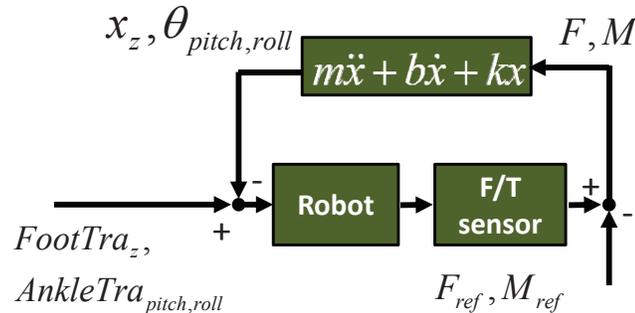
Figure 5.7: Block diagram for compliance control

role of moment compliance is to keep the foot area fully contacting the surface while the force compliance is active. The F/T sensor used in this work has a resolution 10^{-4}Nm. Thus, it is assumed that the foot contacts the ground properly if the moment measurements are zero. Note that there is coupling in sensing force and moments. Thus, the response of moment compliance control must be faster than that of force compliance control.

5.5 Compliance Control Design with a Variable Stiffness

Both feet are fully constrained to the ground in the DSP while the CoM moves from the rear to the front foot. In this transition, the weight loaded on the rear foot decreases while the weight on the front foot increases. The shift in weight results in a kinematic error between the legs because of indeterminate compliance in the mechanical structure. Such kinematic errors in the constrained posture cause over torquing in joints. Under observation, the moments on the foot by such torque become

critical for balance in walking with a larger step.

The indeterminate compliance typically comes from joint actuators, force-torque sensors, and compliant materials like rubber-pads on the feet. It behaves like a high-stiffness non-linear spring depicted in Figure. 5.6 with notation, k_{non}. To minimize the influence of k_{non}, the active compliance is applied to the DSP.

According to the definition of ZMP, the ZMP is the only point where force may act. Thus, the force profile in the DSP can be derived from the reference ZMP trajectory. The ZMP trajectory in y-direction is normalized to a range from 0 to 100 where the ZMP is zero when it is on the center of left foot. On the other hand, it is 100 when it is at the center of the right foot. The force profiles for the right and left foot are obtained in Eq (5.12) and Eq (5.13), respectively,

$$F_{ref}^{RF} = ZMP_{normalized} \times W_t \tag{5.12}$$

$$F_{ref}^{LF} = (100 - ZMP_{normalized}) \times W_t \tag{5.13}$$

where RF and LF denotes the right and left foot.

In particular, the CoM transition in the DSP changes the system dynamics. The impedance control scheme in Figure. 5.7 requires impedance matching which is very cumbersome in this case. Instead of the force tracking control, compliance model with a variable stiffness is built as shown in Eq (5.14).

$$m\ddot{x} + b\dot{x} + k(w)x = F_{meas} \tag{5.14}$$

$k(w)$ is a function of the weight distribution. $k(w)$ for the right leg is defined in Eq (5.15) and for the left leg can be specified in the same manner.
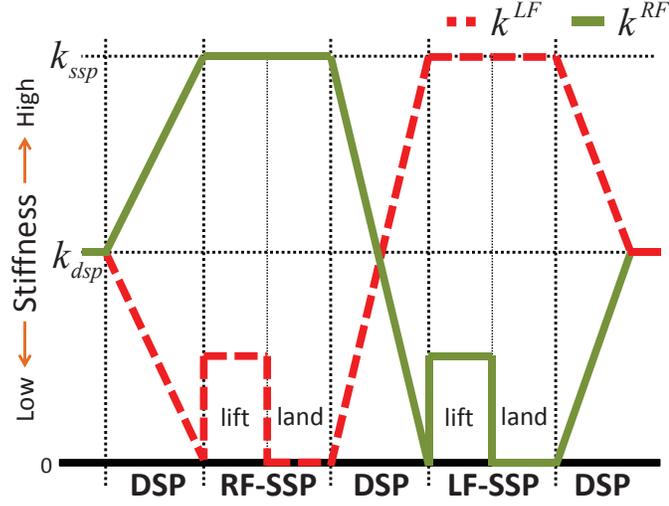
Figure 5.8: Stiffness map for compliance control

$$k(w)^{RF} = k_{ssp}^{RF} F_{ref}^{RF}/W_t \tag{5.15}$$

k_{ssp}^{RF} in Eq (5.15) is a constant stiffness coefficient used for Right-Foot SSP (RF-SSP). The value of k_{ssp}^{RF} is determined empirically.

In Figure 5.8, the stiffness map is depicted phase by phase in walking. Four force compliance models for walk-ready pose, SSP, DSP, and landing for each leg are independently designed and switched at the phase transition. k_{dsp} is a constant stiffness coefficient used for walk-ready posture supported by both legs. k_{ssp} needs to have enough stiffness to support the entire weight of the robot as a single leg. Once k_{ssp} is determined, the variable stiffness in the DSP is obtained from Eq (5.12), Eq (5.13), and Eq (5.15).

Here, the SSP is divided into the foot lifting and landing phases. The position of rear foot in the vertical direction is adjusted for compliance in the DSP. In the foot

lifting phase, such controlled position is eliminated so that the swing foot follows the desired foot trajectory correctly. In the last half of the SSP, the landing compliance control in Eq (5.11) is turned on to prepare the landing at anytime during this phase.

5.6   Observer-Based Damping Control

Humanoids are electro-mechanical systems. As such, they are typically influenced by mechanical uncertainties like vibration, deformation, and compliance. These factors often resonate together and amplify disturbances. In particular, a two-foot or one-foot support posture becomes like an inverted pendulum which has a natural oscillating frequency. Once it resonates to such disturbances, the robot often becomes unstable. Kim et. al. [59] designed a damping controller to minimize the mechanical oscillation for locomotion. This controller is combined with the LQI balance controller discussed in Section 5.1 in order for robust performance against disturbances in manipulation and locomotion.

Based on the dynamics of an inverted pendulum, the control objective is to minimize the mechanical oscillation. The control input is the CoM position and its output is the ZMP. The equation of motion of the LIPM is derived in Eq (3.15). Its state-space representation is shown in Eq (5.16).

$$\frac{d}{dt}\begin{bmatrix} \theta_c \\ \dot{\theta}_c \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} & 0 \end{bmatrix} \begin{bmatrix} \theta_c \\ \dot{\theta}_c \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{m_t l^2} \end{bmatrix} \tau \tag{5.16}$$

where l and m_t are the length from the ankle joint to the CoM location and total mass, respectively. g is the gravitational acceleration and τ is the torque exerted on the ankle joint. θ_c is the current angle of CoM measured from the vertical axis.

Humanoids are typically position-controlled. τ should be represented as a form of joint angles. Here, τ is defined as $K_p(\theta_c - \theta_d)$ where K_p is the mechanical stiffness and
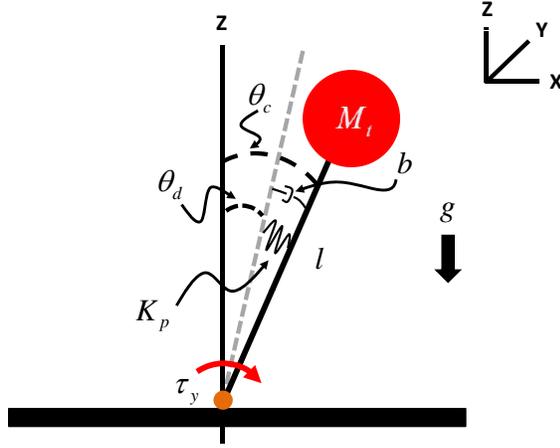
Figure 5.9: The spring-damper inverted pendulum model for a humanoid. θ_c and θ_d denote the current and desired angles, M_t and l are the total mass and length to CoM from the ankle joint, and K_p and b are the mechanical stiffness and compliance, respectively.

θ_d is a desired reference angle. To control the CoM position in Cartesian space, $l\theta_d$ is linearized to ΔCoM_d which is the control input to the system. With the mechanical compliance, b, the LIPM in Eq (5.16) becomes the spring-damper inverted pendulum represented in Eq (5.17) and depicted in Figure 5.9.

$$\frac{d}{dt}\begin{bmatrix}\theta_c \\ \dot{\theta}_c\end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} - \frac{K_p}{m_t l^2} & -\frac{b}{m_t l^2}\end{bmatrix}\begin{bmatrix}\theta_c \\ \dot{\theta}_c\end{bmatrix} + \begin{bmatrix}0 \\ \frac{K_p}{m_t l^3}\end{bmatrix}\Delta CoM_d \qquad (5.17)$$

The open-loop system in Eq (5.17) is the typical form of a second-order system. But, the system has the mechanical stiffness, K_p, and compliance, b, which are inderterminate parameters. These can be obtained from experiments [59]. The second-order system identification method is used to specify these parameters.

The typical form of a second-order transfer function is structured as in Eq (5.18). Its impulse response is shown in Figure 5.10.
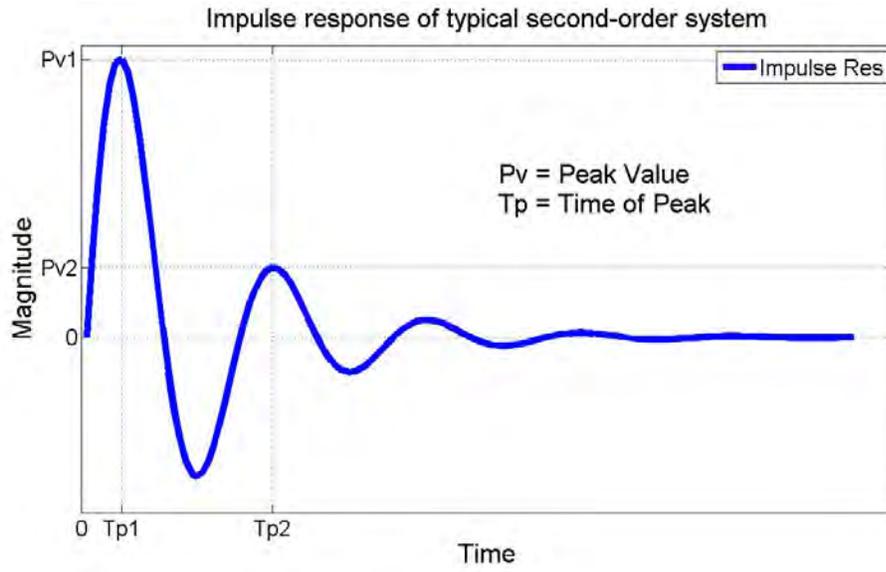
Figure 5.10: Impulse response of a typical second-order system. P_v and T_p denote the peak value and peak time.

$$G(s) = \frac{\alpha \omega_n^2}{s^2 + 2\zeta \omega s + \omega_n^2} \qquad (5.18)$$

where α is the DC-gain. ζ and ω_n denote the damping ratio and natural frequency, respectively.

From the impulse response in Figure 5.10, ζ and ω_n are obtained from formulations in Eq (5.19), Eq (5.20), and Eq (5.21).

$$\omega_d = \frac{2\pi}{T_{p2} - T_{p1}} \qquad (5.19)$$

$$\ln \frac{P_{v1}}{P_{v2}} = \frac{2\pi\zeta}{\sqrt{1 - \zeta^2}} \qquad (5.20)$$

$$\omega_d = \omega_n \sqrt{1 - \zeta^2} \qquad (5.21)$$

where ω_d is the damped natural frequency, and T_p and P_v are the peak time and value in output response.

The parameter comparison between Eq (5.17) and Eq (5.18) yields that $\omega_n = \frac{K_p - mgl}{ml^2}$ and $2\zeta\omega_n = \frac{b}{ml^2}$. Hence, the mechanical stiffness and compliance are calculated from Eq (5.22) and Eq (5.23), respectively.

$$K_p = ml^2(\omega_n^2 + g/l) \tag{5.22}$$

$$b = 2\zeta\omega_n + ml^2 \tag{5.23}$$

The humanoid is modeled as a second-order system about the CoM position displacement and angular displacement at the ankles. Now, this system is transformed to a relationship between the CoM displacement and ZMP because the ZMP is usually measurable in most humanoids.

According to the ZMP definition in Eq (3.7), the ZMP equals to $-\frac{\tau}{m_t g}$. Since the torque is defined as $K_p(\theta_c - \theta_d)$, the output ZMP can be defined as Eq (5.24).

$$ZMP = \begin{bmatrix} \frac{K_p}{m_t g} & 0 \end{bmatrix} \begin{bmatrix} \theta_c \\ \dot{\theta}_c \end{bmatrix} + \frac{-K_p}{m_t g}\theta_d \tag{5.24}$$

With Eq (5.17), the state-space form of the spring-damper inverted pendulum system can be represented as Eq (5.25).

$$\begin{aligned} \dot{\theta}_c &= A\theta_c + Bx_d \\ ZMP &= C\theta_c + DCoM_d \end{aligned} \tag{5.25}$$

$$A = \begin{bmatrix} 0 & 1 \\ \frac{g}{l} - \frac{K_p}{m_t l^2} & -\frac{b}{m_t l^2} \end{bmatrix}, B = \begin{bmatrix} 0 \\ \frac{K_p}{m_t l^3} \end{bmatrix}$$

$$C = \begin{bmatrix} \frac{K_p}{m_t g} & 0 \end{bmatrix}, D = \frac{-K_p}{m_t gl}$$

The transfer function from the CoM displacement to ZMP measurement is derived
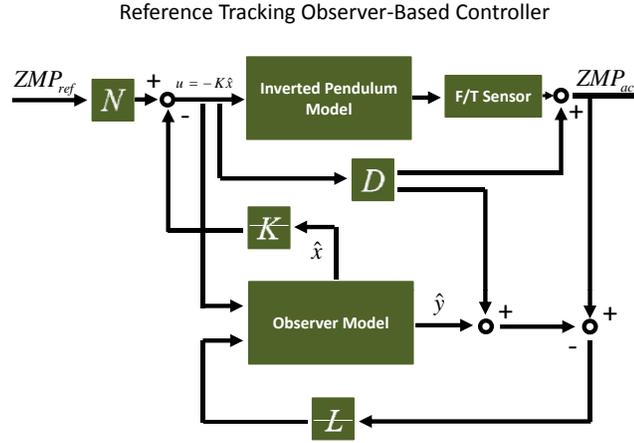
Figure 5.11: Block diagram of the real-time ZMP compensator using observer-control law. K and L are the regulator and observer gains, respectively.

in Eq (5.26).

$$ZMP = \frac{K_p}{mgl}\left(\frac{-s^2 - \frac{b}{ml^2}s + \frac{g}{l}}{s^2 + \frac{b}{ml^2}s + \frac{K_p - mgl}{ml^2}}\right)CoM_d \tag{5.26}$$

The transfer function has a complex pole located near imaginary axis and a zero in the RHP (Right-Half Plane). The zero in RHP often make the design problem cumbersome because this zero pulls root loci to RHP. To implement a full-state feedback control, the observer is designed to estimate the states from the ZMP measurements. The controllability matrix of Eq (5.25), ($\begin{bmatrix} C \\ CA \end{bmatrix}$), is of full rank and thus the system is observable. With the classical observer-control law in Eq (5.27), the ZMP compensator has a control flow shown in Figure 5.11.

$$\dot{\hat{\theta}}_c = (A - BK - LC + LDK)\hat{\theta}_c + Ly$$
$$u = -K\hat{\theta}_c \tag{5.27}$$

where $\hat{\dot{\theta}}_c$ is the state vector estimated by Eq (5.27). K and L are the regulator and observer gains, respectively.

According to the classic observer-control law, the regulator determines the system performance while the observer estimates the states of the system. The desired performance in this ZMP compensator is specified with %Over-Shoot (%OS) and peak time (T_p). The desired damping ratio and natural frequency now can be calculated from Eq (5.28) and Eq (5.29), respectively.

$$\zeta_{desired} = -\frac{\log \%OS}{\sqrt{\pi^2 + \log \%OS^2}} \tag{5.28}$$

$$\omega_{desired} = \frac{\pi}{T_p \sqrt{1 - \zeta_{desired}^2}} \tag{5.29}$$

Using the pole-placement method, the regulator gain, K, is calculated. The observer gain, L, must be selected to allow the states to converge 4 to 10 times faster than the system response.

5.7 Control Performance Evaluation

LQI-based Balance Control

The real-time controllers proposed and designed in this chapter were tested and evaluated on a full-sized humanoid, the DRC-Hubo. DRC-Hubo was designed in 2013 by KAIST in South Korea in order to perform tasks in real world environments (see Appendix A).

The LQI balance controller was implemented. Figure 5.12 depicts the performance evaluation about the external disturbances. A tool box with 11.8lb of weight was affixed arbitrarily to the DRC-Hubo. The force and moment were measured at a rate of 200Hz. The controller moved the pelvis position to counterbalance the weight.
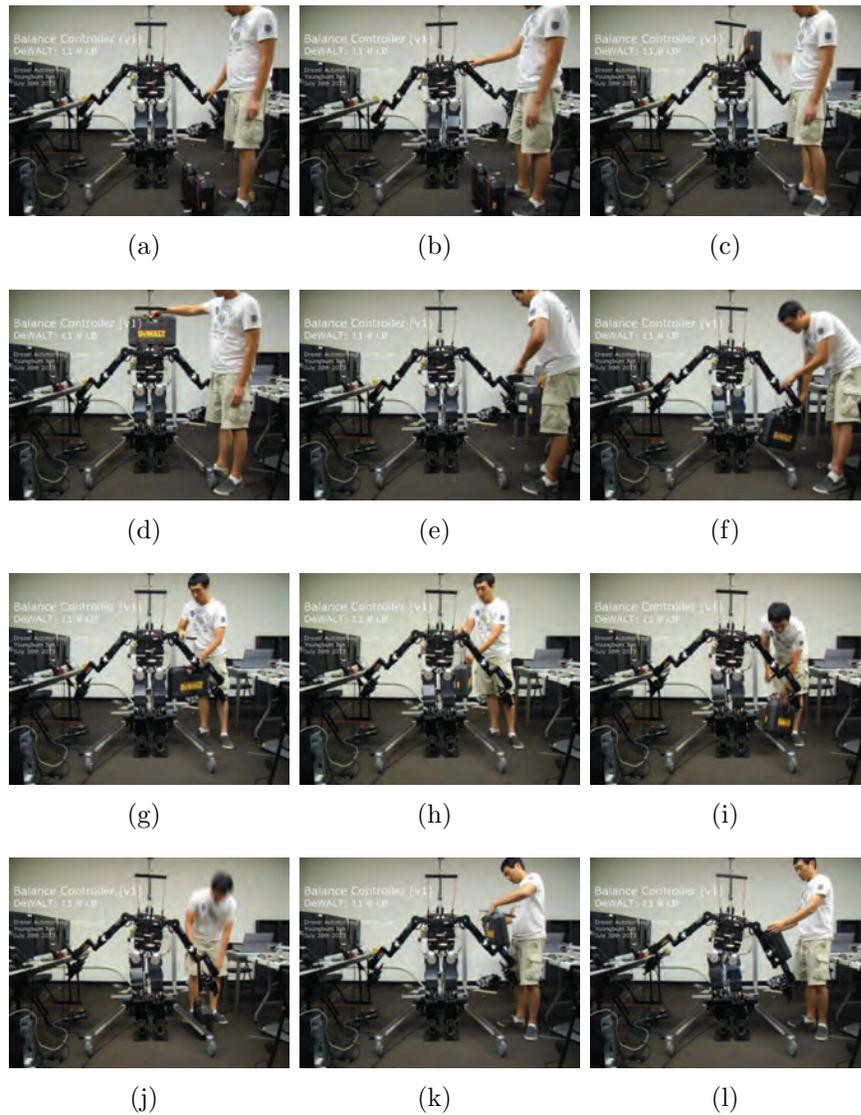
Figure 5.12: Performance evaluation of LQI balance controller against interactive forces. A tool box weighing $11.8lb$ ($5.35kg$) is loaded onto the DRC-Hubo.

(a) 0.0s   (b) 0.5s   (c) 1.0s

(d) 1.5s   (e) 2.0s   (f) 2.5s

(g) 3.0s   (h) 3.5s   (i) 4.0s
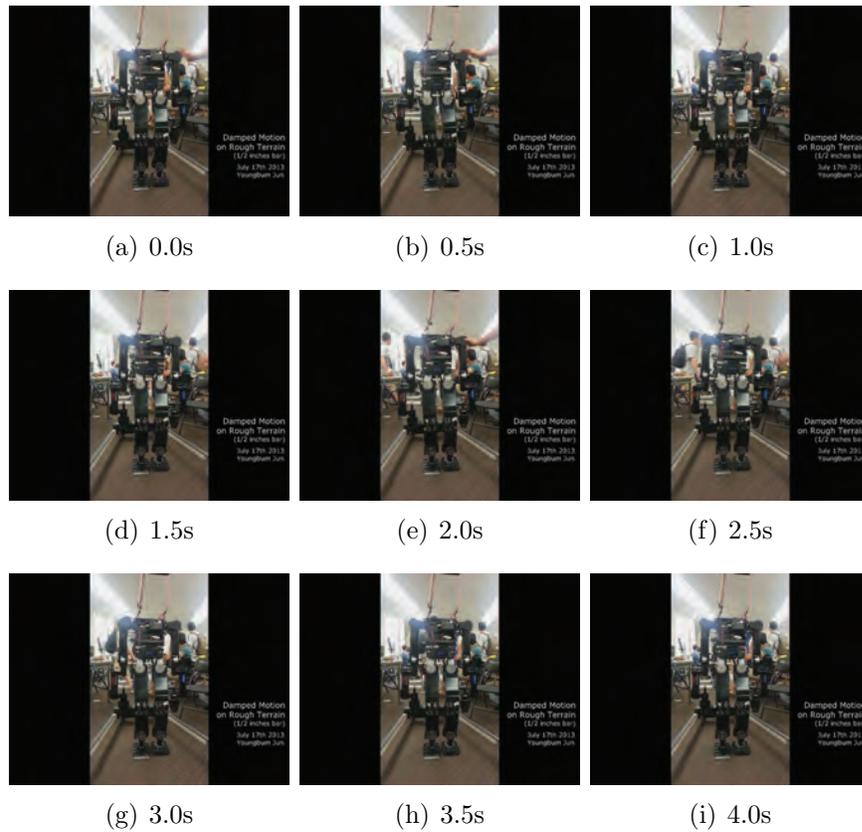
Figure 5.13: Disturbance rejection and minimum oscillation through the damping controller. DRC-Hubo standing on a $\frac{1}{2}$ in. thick aluminum bar is disturbed by external forces.

This balance controller was designed using a double inverted pendulum model.

Observer-based Damping Control

Observer-based damping controller was realized on DRC-Hubo. Figure 5.13 is a snapshot of the control performance against the external disturbances. A human pushed the robot standing on a $\frac{1}{2}$ in. plate with an arbitrary amount of forces. The robot recovered its original posture within 1.0 second. Figure 5.14 shows plots of the ZMP measurements compared with the desired performance. From the undamped response (blue dotted line), the dynamic model was identified. The full-state feedback
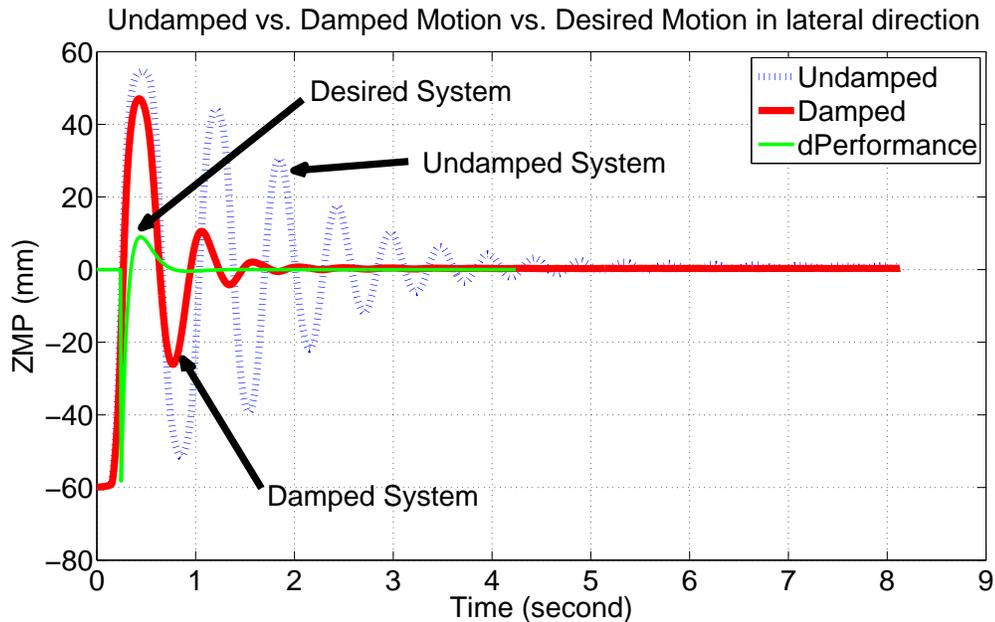
Figure 5.14: Performance analysis between undamped, damped, and desired system.

controller (green light line) was designed via an observer in order to have a $\%OS$ less than 2% and peak time of 0.6 seconds from a unit step input. The experimental data, outlined in bold red, shows a significantly decreased mechanical oscillation and follows the desired control specification.

Real-Time Gait Planner with Compliance Control

A real-time biped gait planner with landing compliance controller was implemented in DRC-Hubo. Figure 5.15 is a snapshot of DRC-Hubo dynamically walking on arbitrary terrains. DRC-Hubo detected the landing earlier or later according to the force measurements from the F/T sensors located in both ankle joints. A real-time gait planner generated the gait patterns while the virtual compliance absorbed the impact forces during foot landing.

Through the performance evaluations of real-time feedback controllers, the stability and adaptability of a full-sized humanoid to arbitrary environments were fully

Figure 5.15: Real-time biped gait planner with landing compliance control.

demonstrated. The humanoid is able to handle the interactive forces, to reject the disturbances, and to locomote on unstructured terrains.

Biped Locomotion on Rough Terrains with Real-Time Controllers

All real-time controllers were integrated with gait planner introduced in Chapter 4. The ideas were fully verified and validated via biped locomotion on uneven and rough terrains. Figure 5.16 is a snapshot of DRC-Hubo stepping over a 4×4 wood bar. Figure 5.17 depicts an ability of the humanoid to climb up/down a cinder block, 6 inches height. Figure 5.18 and Figure 5.19 show the humanoid walking on arbitrary even rough terrains and ramps, respectively. These experiments demonstrate robustness and adaptability of the real-time controllers designed in this chapter.
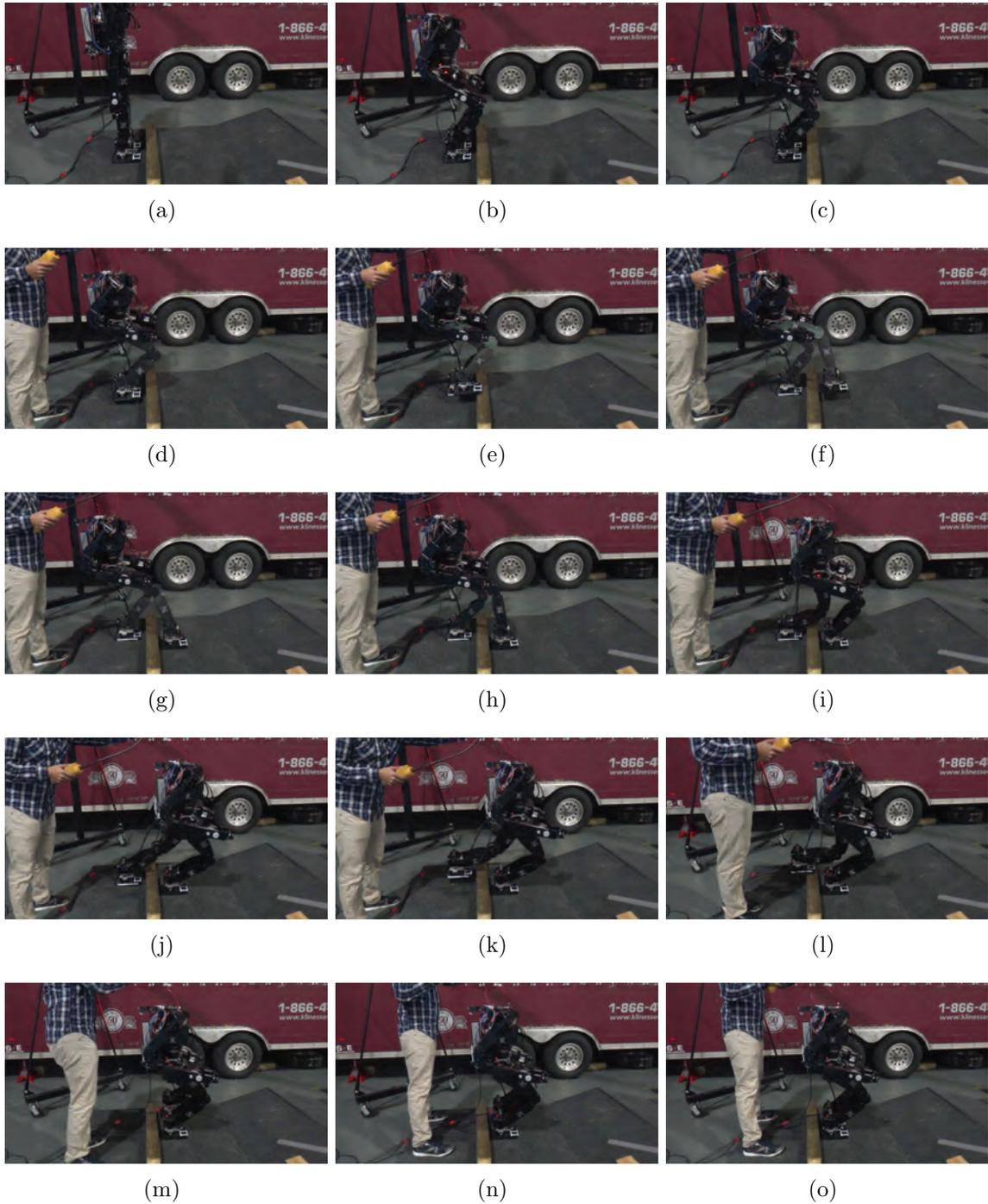
Figure 5.16: DRC-Hubo stepping over a 4×4 obstacle.

Figure 5.17: DRC-Hubo walking through an obstacle - a cinder block, 6 in. tall and 10 in. wide.
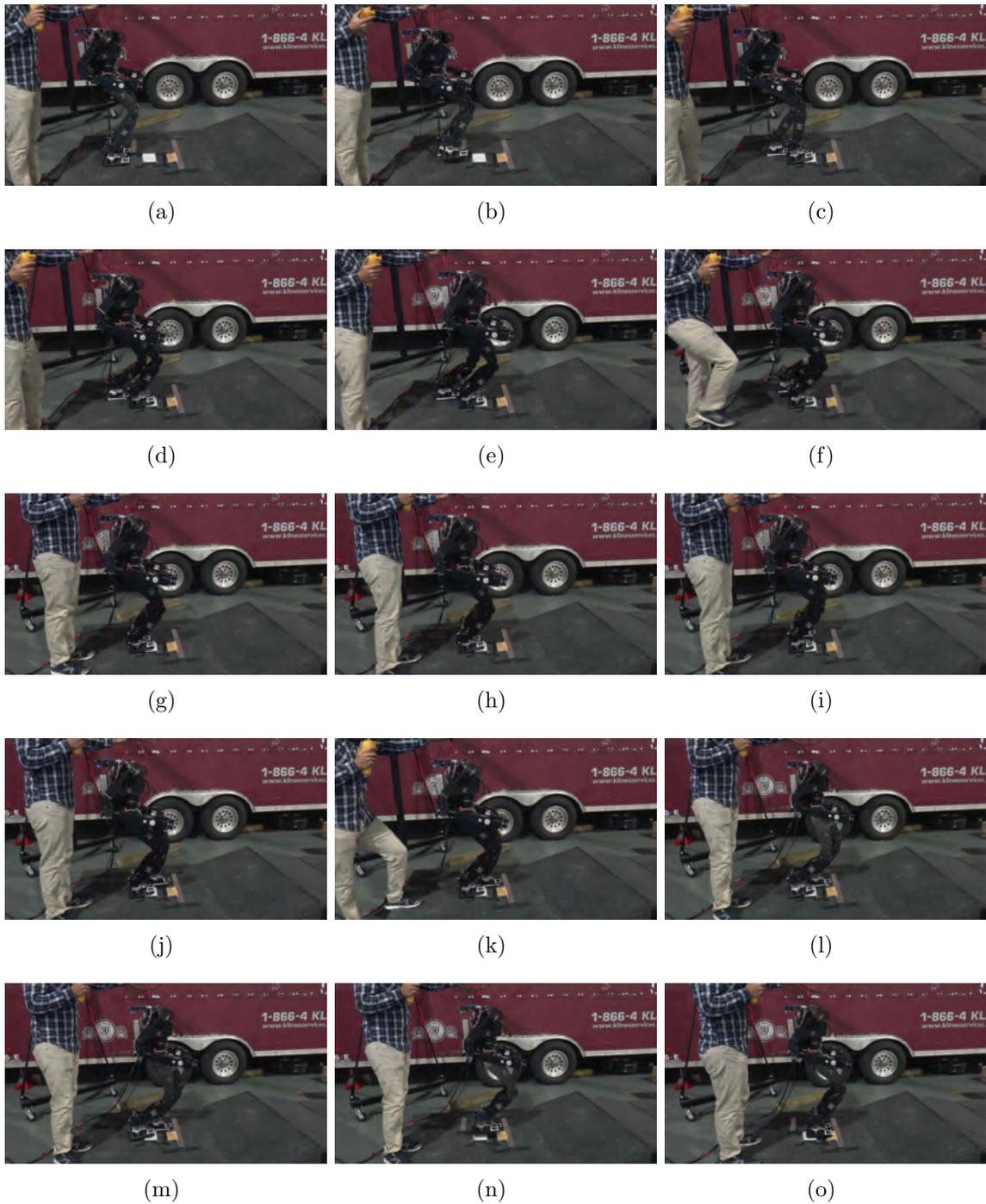
Figure 5.18: DRC-Hubo walking on an arbitrary rough terrain.

(a)　　　　　　　　　(b)　　　　　　　　　(c)

(d)　　　　　　　　　(e)　　　　　　　　　(f)

(g)　　　　　　　　　(h)　　　　　　　　　(i)

(j)　　　　　　　　　(k)　　　　　　　　　(l)

(m)　　　　　　　　　(n)　　　　　　　　　(o)

Figure 5.19: DRC-Hubo walking on 15deg inclinded ramp with an arbitrarily rough surface.

6. WHOLE-BODY MANIPULATION

Current state-of-the-art humanoids have demonstrated advanced capabilities in manipulating tools and objects. With high level sensing, even in increasingly complex environments, these robots are able to carry heavy boxes, hammer nails, open doors, and even drive cars. However, these applications have not been demonstrated robustly in the presence of real-world phenomena like unexpected terrain properties, wind forces, adverse weather conditions, and other difficult to model perturbations. To extend these advanced capabilities to practical, real world situations, such interactions should be addressed.

In this chapter, humanoid whole-body manipulation is demonstrated. The whole body motion control, Operational Space Formulation (OSF) [97], is implemented. Through the pushing task, the dynamic model and relationship between the postures and reactionary forces are analyzed. For stable and robust manipulation in outdoor environments, a posture controller is designed. These works are fully tested and evaluated experimentally. An autonomous peg-in-hole task is demonstrated to verify and validate the approaches.

6.1   Double inverted pendulum model

Harada et. al. [27] derived the linear relationship between the Zero-Moment Point and reaction force based on an inverted pendulum model, [49]. To better model the effects of the center of mass location compared to the foot and hand contacts of the robot, the ZMP model was expanded to work with the DIP model. This model considers the additional effects of the upper body location on the support polygon.

The Double Inverted Pendulum Model (DIPM) models the upper body as another
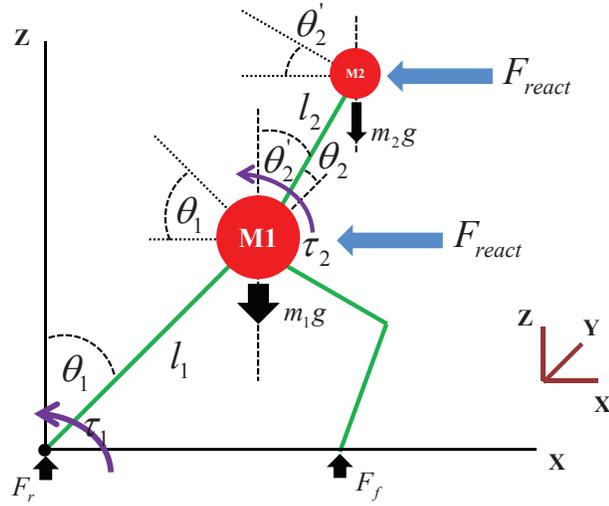
Figure 6.1: A force model of the Double Inverted Pendulum in a static posture

inverted pendulum [82]. Figure 6.1 represents the 2-dimensional kinematics and dynamics of the DIPM in the feet-apart posture. The origin in Figure 6.1 represents the ankle joint in the rear leg. m_1 and m_2 are the lower body point mass and upper body point mass, respectively. In this paper, the author assumes that m_1 and m_2 are located in the hip and chest, respectively, and the lower body link, l_1, and upper body link, l_2, are rotating with respect to the ankle and hip joints, respectively, in the humanoid. τ_1 and τ_2 denote the torque acting on the ankle joint of the rear leg and the hip joint. The dynamic equation of the DIPM without the external force is Eq (6.1).

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = M(\theta)\ddot{\theta} + V(\theta, \dot{\theta}) + G(\theta) \tag{6.1}$$

Where M is a 2×2 inertial matrix, V is a 2×2 matrix of centrifugal and Coriolis

components, and G is 2×2 matrix containing the gravitational constant.

In the static posture, the acceleration and velocity of the upper limbs and lower limb can be ignored. Accordingly, Eq (6.1) relies only on the gravitational constant as shown in Eq (6.2).

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} -(m_1 + m_2)l_1 g \sin(\theta_1) - m_2 l_2 g \sin(\theta_1 + \theta_2) \\ -m_2 l_2 g \sin(\theta_1 + \theta_2) \end{bmatrix} \tag{6.2}$$

6.2 Analysis of Postures and Reactionary Force

The ZMP equation is simply defined as Eq (6.3). Thus, the ZMP in the forward direction, X, is as shown in Eq (6.4).

$$\frac{-\tau_1}{F_z} = ZMP = \frac{-\tau_1}{(m_1 + m_2)g} \tag{6.3}$$

Where F_z is the ground reaction force that equals to $(m_1 + m_2)g$.

$$ZMP_x = l_1 \sin(\theta_1) + \frac{m_2}{m_1 + m_2} l_2 \sin(\theta_1 + \theta_2) \tag{6.4}$$

With the external force, τ_1, which is the joint torque on the ankle of the rear leg, is defined as in Eq (6.2).

$$\tau_1 = -(m_1 + m_2)l_1 g \sin(\theta_1) - m_2 l_2 g \sin(\theta_1 + \theta_2) + \tau_{react}$$

$$\text{Where } \tau_{react} = F_{react}(l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)) \tag{6.5}$$

With Eq (6.3), the important relationship between the horizontal reaction force in a given posture is derived in Eq (6.6).

$$\frac{1}{a} \triangle ZMP = \triangle F_{react} \tag{6.6}$$

$$\text{Where } a = \frac{l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)}{(m_1 + m_2)g}$$

Since $\theta_1 + \theta_2$ are numeric values that can be obtained from a given posture. Thus, the relationship between the reaction force and ZMP is linear. It is the same result as in [27].

Based on Eq (6.6), the ZMP is moving from the initial position to the rear support foot when the pushing force is applying to the target object. Assume that the friction is enough for the humanoid to tip over, the humanoid starts to tip while the ZMP is passing through the location of the rear foot. Mathematically, such conditions can be defined as $\tau_1 = 0$. Thus, the pushing force of a given posture can be calculated using Eq (6.3) and Eq (6.7).

$$F_{max} = \frac{(m_1 + m_2)l_1 g \sin(\theta_1) + m_2 l_2 g \sin(\theta_1 + \theta_2)}{l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2)} \tag{6.7}$$

Where F_{max} is the force limitation that the humanoid can support. As long as there is no vertical force pressing the humanoid down, Eq (6.7) is valid. Thus, the force controller to manipulate the target object should be designed within the force limitations, and its performance should not exceed that of the maximum force so as to maintain balance while performing a pushing task.

In previous works, the pushing force has been controlled by horizontally changing the position of CoM while the upper body is upright with respect to the ground in order to keep the height of the contact point between the arms and object. With this idea, the author controls the pushing force using both lower and upper body simultaneously while maintaining the height of the contact point.

Assume that the humanoid is at its initial pose, and the height of the target object is specified as in Eq (6.8).

$$P_h = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) = const. \tag{6.8}$$

Where P_h is the pushing height. All possible postures with the constant height will satisfy the condition shown in Eq (6.9).

$$P_h = l_1 \sin(\theta'_1) + l_2 \sin(\theta'_1 + \theta'_2) \tag{6.9}$$

Where θ'_1 and θ'_2 are the angle of the lower body and upper body, respectively. Using Eq (6.9), the posture that exerts the maximum pushing force in a given initial posture and height of the object can be obtained by Eq (6.3).

$$max[\theta_1, \theta_2] =$$
$$max(\frac{(m_1 + m_2)l_1 g \sin(\theta'_1) + m_2 l_2 g \sin(\theta'_1 + \theta'_2)}{l_1 \cos(\theta'_1) + l_2 \cos(\theta'_1 + \theta'_2)}) \tag{6.10}$$

6.3   Experimental Pushing Force Analysis

Simulation

Figure 6.2 represents the simulation data of Eq (6.7). The five dotted lines represent ZMP trajectories, while the five bold lines show pushing force limits (Max. Force) that correspond to the ZMP. The line at the top in the dotted group is the ZMP plot based on the lower body angle that varies from 0.45 to 0.7 rad and upper body angle that equals 20deg. The force plot corresponding to that ZMP plot is shown at the top in the bold group. In the same manner, each dotted line for ZMP from the bottom to the top shows the upper limb angle from -20, -10, 0, 10, and
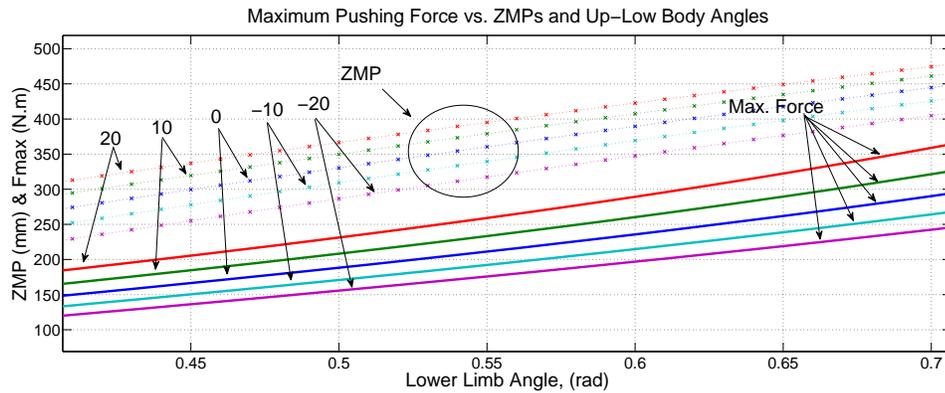
Figure 6.2: Maximum pushing force vs. ZMP and Postures
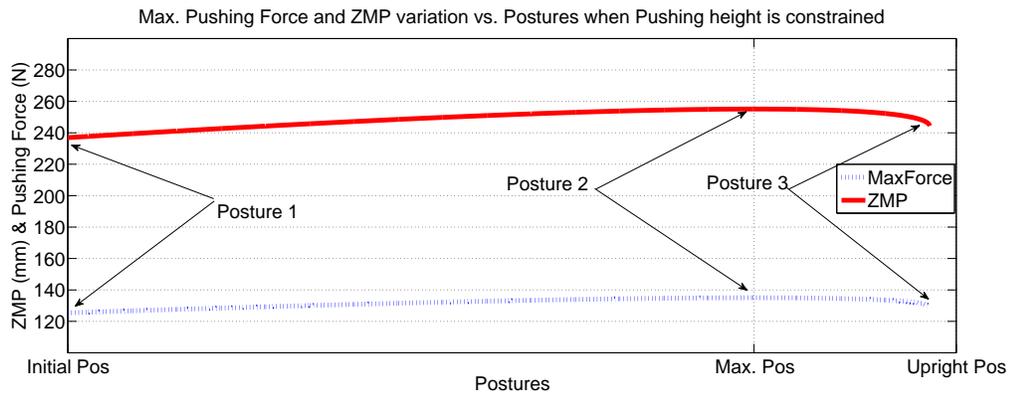
Figure 6.3: Maximum pushing force and ZMP variation from posture 1 to posture 3

20deg and corresponds to each bold line for force limitation from the bottom to the top, respectively.

There are two facts that must be emphasized. First, the force limitation of each posture is unique. Second, even if some postures have the same ZMP values, the force limitations are different between them because of the different upper body angles. It gives an idea that there is a unique posture that can produce the maximum pushing force in accordance with the height of the contact point.

In Figure 6.3 shows the pushing force and ZMP variation in different postures

(a) Initial posture   (b) Maximum pushing posture (c) Upper-body upright posture
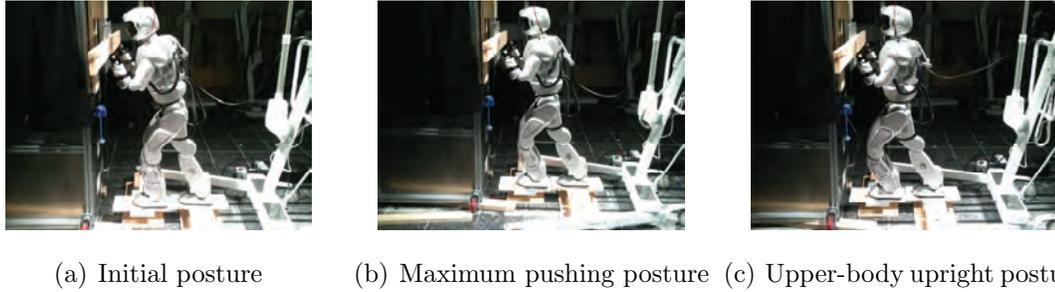
Figure 6.4: Experimental evaluation of the relationship between pushing force and postures.

when the contact point is specified. To clarify, the arrow named Posture 1 in Figure 6.3 indicates the maximum pushing force (red and bold line) and ZMP (blue and dot line) of the robot posing in Figure 6.4(a). The arrow named Posture 2 and Posture 3 also point to both maximum pushing force and ZMP, and correspond to postures in Figure 6.4(b) and Figure 6.4(c), respectively. Figure 6.4(a) is the initial posture. Figure 6.4(b) and Figure 6.4(c) are the posture of the maximum pushing force and the upright posture obtained from Eq (6.3), respectively. Figure 6.4(a) is the initial posture when $\theta_1 = 20 \deg$ and $\theta_2 = 0 \deg$. Posture in Figure 6.4(b) shows the maximum pushing force posture when $\theta_1 = 24.63 \deg$ and $\theta_2 = -15.21 \deg$ and Figure 6.4(c) is the upper body upright posture when $\theta_1 = 25.81 \deg$ and $\theta_2 = -25.60 \deg$.

In the simulation result in Figure 6.3, there is a unique posture that can generate the maximum pushing force without changing the height of the contact point. Beyond our simulation, the humanoid can achieve the maximum pushing posture using lower and upper body angles.

Experimental Result

Figure 6.4 shows the experimental set up. The reaction force and ground reaction force on each foot are recorded while Hubo+ is pushing the wall. Figure 6.5 represents
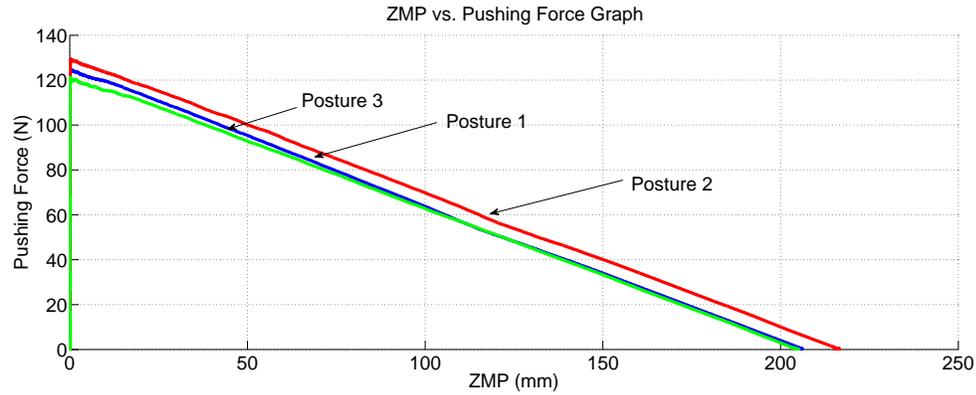
Figure 6.5: ZMP vs. Pushing force

the reaction force vs. ZMP plot of posture 1, 2, and 3. The experimental data in Figure 6.5 fits expectations based on the results of Figure 6.3.

Figure 6.6 describes the comparison between the simulation data in Figure 6.3 and the experimental data. Figure 6.6(a), Figure 6.6(b), and Figure 6.6(c) depict postures 1, 2, and 3, respectively. The bold blue lines in each plot represent the simulation data using DIP model and the red dotted lines in each graph denote the experimental data from the Hubo.

6.4 Case Study: Manipulation on Unstructured Terrains

Posture is an important factor in improving the physical capability of humanoids for a given task. Lowering the body increases the stability to external force perturbations. Keeping the feet further apart also increases the robot's ability to resist external forces, which maximizes the robots ability to apply pushing and pulling forces during manipulation tasks[42]. In planning and control, we can use a mathematic model to chose a desired posture which minimizes uncertainties in sensing and performance. In this work, we address the question of what criterion we should use to find postures that are robust and stable.
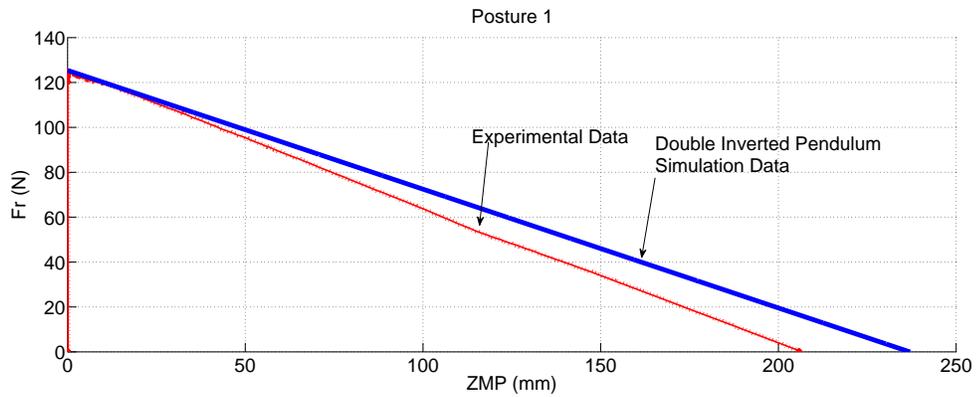
(a) ZMP vs. Pushing force for Initial posture
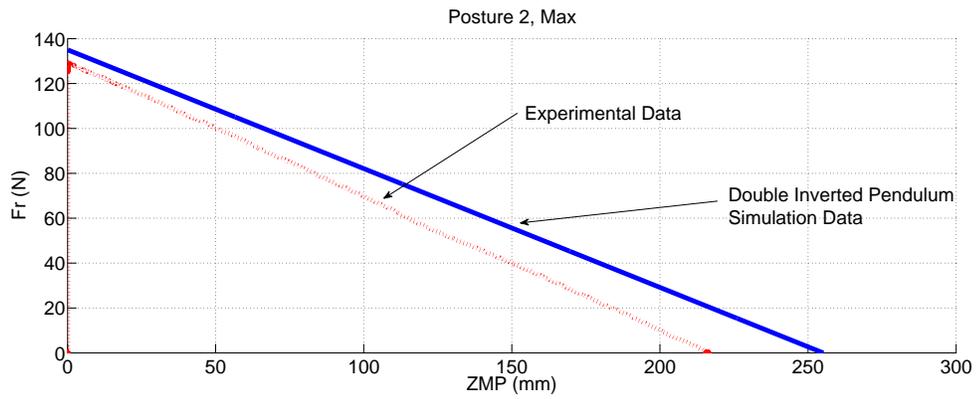
(b) ZMP vs. Pushing force for Maximum pushing posture
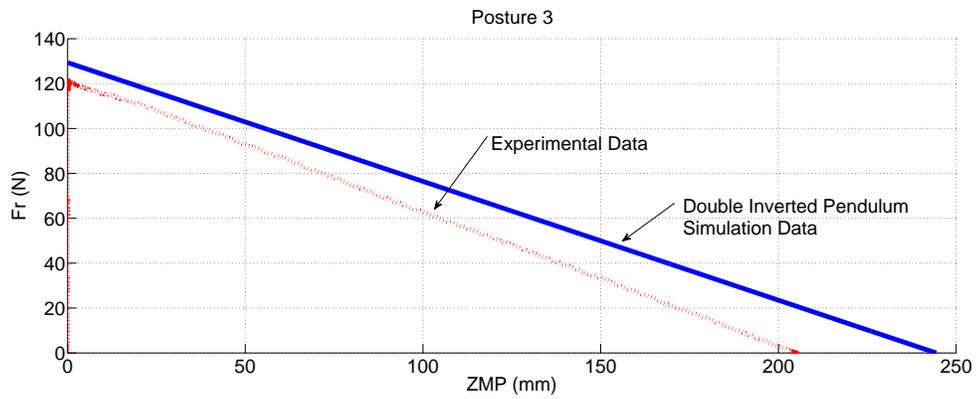
(c) ZMP vs. Pushing force for Upper body upright posture

Figure 6.6: Experimental evaluation of the relationship between pushing force and postures.

Figure 6.7: (a) A possible posture where the feet do not fully contact the ground. (b) The posture suitable to perform a manipulation task in place.

Figure 6.8: (a) A posture that yields the actual ZMP off from estimated ZMP. (b) The posture more robust than in (a) for manipulation in place
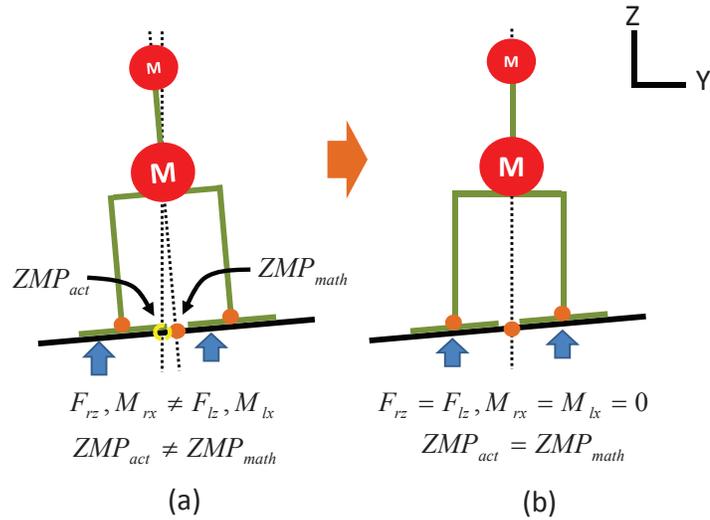
Figure 6.9: (a) A posture on a slight incline. (b) The posture that has length of legs and orientation adjustment.

Humanoids are typically position controlled and lack compliance. However, because there are many joints and the robots are relatively heavy, planned motions still often result in kinematic errors that make the desired posture impossible to reach and difficult to sense. Thus, the performance of controllers designed purely with an analytical kinematic model is generally limited. These factors often result in unbalanced postures on rough terrain shown in Figure 6.7(a), Figure 6.8(a), and Figure 6.9(a). In these postures, the robot may not accurately sense it's current status with respect to relative position and orientation, and stability. Such errors significantly influence the performance of motion and control.

In the figures presented, $RFpos$ and $LFpos$ denote right and left foot positions, respectively. F and M are the force and moment acting on the foot, respectively. the first characters in the subscripts, r and l, stand for right and left, respectively, and the second characters, x and z, represent the axis in a local coordinate system originated at the center of each foot. ZMP_{act} and ZMP_{math} are the Zero-Moment

Point measured from sensors and calculated from a kinematic model, respectively.

Figure 6.7(a) illustrates the robot standing on curved terrain, where the feet do not fully contact the ground. This may occur either due to kinematic errors or due to limited performance in compliance control. To describe how these errors may lead to discrepancies between the expected Zero-Moment point and the actual Zero-Moment point, let us assume that the same amount of force and moment are acting on the same point on the feet. The calculation for the actual ZMP through sensory measurements with a typical beam-strain gauge Force-Torque sensor (F/T sensor) is shown in Eq (6.11).

$$ZMP_{x,y} = \frac{(P^{RF} + M^{RF}) + (P^{LF} + M^{LF})}{F_{rz} + F_{lz}} \tag{6.11}$$

where the superscripts RF and LF denote right and left foot, respectively. P is the position in the local frame. Eq (6.11) is derived and holds the condition, Eq (6.12),

$$(P^{R}_{react} - ZMP) \times F_{rz} + (P^{L}_{react} - ZMP) \times F_{lz} = 0 \tag{6.12}$$

where P^{R}_{react} and P^{L}_{react} are the positions where the reaction force is acting in right and left side in the local frame, respectively.

From Eq (6.11), the actual ZMPs in x and y directions are zero and thus located at the origin. This means that the robot is fully stable. However, in practice one often sees that there is less robustness against external forces(or reactionary force in manipulation) in the y-axis when there is no compliance because of the stiffness of the controller. Forces in that direction result in oscillations back and forth. This often makes the balance controller unstable. In Figure 6.7(b), we show how the posture can be adjusted to avoid such issues.

In Figure 6.8(a), a posture on a gravelly field is depicted. Assume that the mag-

nitude of vertical reaction force on the right and left feet are equal, ($F_{rz} = F_{lz}$), but an unbalanced moment is acting on the right foot, ($M_{rx} \neq M_{lx}$). Using Eq (6.11), we see the actual ZMP produces a non-zero value. But, the estimated ZMP , derived from the kinematic model, is located in the origin because the projection of the Center-of-Mass (CoM) position is on the origin. If the desired ZMP is the origin, a simple feedback controller for balance will move the CoM to compensate the ZMP error. The result is that the robot lost its footing. To prevent this behavior, the posture in Figure 6.8(a) needs to be corrected to that in Figure 6.8(b).

A posture on a slight slope is shown in Figure 6.9(a). The robot is kinematically standing upright while it is actually leaning. The magnitude of moment and force acting on the feet is different, ($M_{rx} \neq M_{lx}$ and $F_{rz} \neq F_{lz}$). Assume that the feet fully contacts to the ground. The actual ZMP from Eq (6.11) is not located on the estimated ZMP. Without adjusting the extension of the legs, the feedback ZMP controller running cannot compensate for the error in the orientation of the robot. The posture should instead be adjusted to that in Figure. 6.9(b).

For minimizing such phenomena in manipulation on rough terrain, we suggest that the posture must be corrected before operating the motion and controller. We propose three conditions for robust postures for humanoid manipulation.

1. The projection of the CoM is located in the sole of convex hull and coincided with the measured ZMP.

2. The mass is equally distributed to both feet

3. No moment acts on each foot

Hereafter, we will refer to these as the robust posture conditions.

6.5   Cascade Robust Posture Control (CRPC) for Manipulation on Unstructured Terrains

The concept of a posture controller was introduced by Kim et. al [57]. It was used for sensory calibration of a humanoid's reference posture. However, the work does not consider the ground roughness and coupling in sensing and motion. The controller designed in this paper overcomes such limitations.

To design a controller for a posture that satisfies the robust posture conditions, the author assumes that humanoids equip two sensors. One sensor can measure the force and moments acting on both feet and the other can sense the angular displacements of the body. With two sensory measurements, four controllers designed here operate in a sequential way in order to avoid the coupling in sensing and motion.

Upper Body Orientation Controller

The upper body posture should be adjustable depending on the manipulation tasks. The goal of this controller is to control the orientation of the upper body to settle at the desired values.

Let us assume that the current upper body orientations for roll, pitch, and yaw are measurable. With the desired orientation values, the control input is simply designed with a proportional gain, K_p, Eq (6.13).

$$\theta_{r,p,y} = K_p(\theta_{r,p,y}^{ref} - \theta_{r,p,y}^{act}) \tag{6.13}$$

where $\theta_{r,p,y}^{ref}$ and $\theta_{r,p,y}^{act}$ are the reference orientations and actual measurements of the upper body, respectively.

More sophisticated controllers can be designed via PID control laws.

ZMP Difference Controller

According to Eq (6.11) and Eq (6.12), moments significantly perturb the ZMP. To satisfy the robust posture conditions, the moments acting on the feet have to be eliminated. The ankle joints are the control objective. The goal of the controller is to provide compliance to reduce the applied moments. Simply, the moments on each foot can be independently controlled. However, it often results in controller instabilities. Once the same non-zero moments are measured from the feet, the joints move in the same direction until the sensed moment becomes zero. Two control steps are used to avoid controller divergence. Firstly, one of foot is adjusted while the other keeps the joint values. Once the controlled foot has the same moment as the other, the ZMP controller introduced in Section 6.5 runs. This strategy removes the moments on the feet at the same time. For the first step, the ZMP value is used to control the foot because the moment sensors are typically very sensitive. The measurement has a lot of noise and variations. The ZMP is a ratio of the moment over vertical force. The ZMP for the right and left foot can be calculated separately. The same ZMP values on the feet means that both ankle joints have the same angle.

Let us assume that the force and torque are measurable. Then, the ZMP difference controller with a proportional gain, K_p, is shown in Eq (6.14) for roll at the ankle joint.

$$\theta_{roll}^{RA} = -K_p\left(\frac{M_{rx}}{F_{rz}} - \frac{M_{lx}}{F_{lz}}\right) \tag{6.14}$$

where θ_{roll}^{RA} denotes the control input for the ankle roll. $\frac{M_{rx}}{F_{rz}}$ and $\frac{M_{lx}}{F_{lz}}$ are the ZMP from measurements of the right and left foot, respectively. The ZMP controller for the pitch angle can be obtained in the same manner as Eq (6.14).

Mass Distributor

Typically, the magnitude of the vertical force acting on the feet is different from each other. The reasons can be from mechanical uncertainties and ground roughness. This controller has a role to equally distribute the total weight to each foot. One way to adjust the force on the foot is to adjust the length of the leg. Since the force values measured from the feet are coupled, bending a leg decreases the force on the foot while the force in the other foot increases, and vice-versa. Thus, the length of a leg is controlled at a time. The mass distributor with a proportional gain, K_p, is designed as Eq (6.15).

$$l_r = -K_p(F_{rz} - F_{lz}) \tag{6.15}$$

where l is the length adjustment and the subscript r is right. The adjustment is added to the current foot position in the vertical direction. The left leg control takes the opposite sign of Eq (6.15). The force sensor is also very sensitive. A large value for K_p often results an unstable motion when the controller is active. K_p is the tuning gain but is typically very small to avoid such instability in control.

ZMP Controller

The ZMP controller controls the ZMP. From Eq (6.11), the actual ZMP is obtained. The first criterion in the robust posture conditions requires the ZMP located at the projection of the CoM position. Through cascade control scheme introduced in Section. 6.5, the step for running this controller already satisfies the first criteria. Therefore, the ZMP can be adjusted by the CoM displacement. With a proportional controller with a gain, K_p, the control input for the CoM displacement is obtained as Eq (6.16).
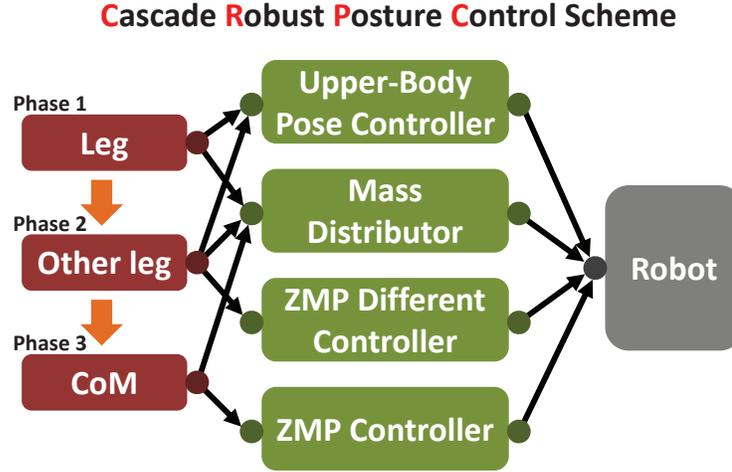
Figure 6.10: Overall control sequence in Cascade Robust Posture Control

$$CoM_{x,y} = K_p(ZMP^{ref}_{x,y} - ZMP^{act}_{x,y}) \qquad (6.16)$$

where the subscript x, y denotes the position in horizontal plane. ZMP_{ref} and ZMP_{act} denote the desired and actual ZMP calculated from Eq (6.11).

Note that real-time Inverse Kinematics are running while such controllers are working.

Cascade Robust Posture Control Scheme

Having robust posture requires that the center of pressure is located within the sole of convex hull, gravitational force exerted on both feet are equal, and no moment acts on either foot. To satisfy such conditions, four independent controllers are designed as previously detailed. Such controllers need the upper-body orientation and force-torque data which is assumed to be measurable. In fact, the control targets require more inputs than the total number of sensors and are also dynamically coupled.

To overcome such difficulties, the author proposes a cascade control scheme named Cascade Robust Posture Control (CRPC) to achieve robust posture in manipulation.

Figure 6.10 shows the overall control sequence in CRPC. In this figure, there are three phases which are defined as Leg, Other Leg, and CoM. In the first phase, it needs to be determined which leg is controlled. As F_z, the measured reaction force, from each foot compared, the leg which has more value in F_z is selected; the leg smaller F_z can cause the kinematic limitation in the length of the leg during control. Then, the upper-body posture controller and mass distributor explained in the previous section run simultaneously. The upper body orientation is controlled based on the reference input and current measured orientation, and the leg chosen is lifted up until F_z from both feet becomes equal. The changes of the upper body orientation and length of the leg are used for Inverse Kinematics (IK) running at every single time to solve all joint angles in lower body. Note that the posture of the upper body influences F_z measured on both feet. Improper gains can cause marginally stable motion in the leg selected.

As a result, the upper body orientation settles at the reference and the weight is equally distributed. Now, the leg not selected for Phase 1 is controlled in Phase 2. The control objective in this phase is to achieve both feet to have the same angles (foot posture). The ZMP in X and Y on each foot is calculated separately, and the joint angles of foot in the leg controlled now is adjusted until the ZMP in X and Y for the currently controlled foot becomes the same ZMP in the other foot. In other words, only one foot is controlled to have the same ZMP on the other foot. When the foot is rotated, F_z on both feet is changed and thus, the mass distributor controls the leg used in Phase 2 to maintain the weight distribution equally. At the same time, the upper body orientation affected by such changes above is kept at the reference. The IK solves all joint angles in the lower body based on such adjustments.

Once the feet have the same posture, the mass is equally distributed and if the upper-body orientation is still at the reference, then the CoM position is adjusted by the ZMP controller. Currently, if the feet have the same ZMP, then the moments acting on each foot are same because F_z on each foot is the same. Based on the total ZMP obtained from Eq (6.11), the CoM position in X and Y direction is controlled until the total ZMP becomes zero in the X and Y directions. In this case, the mass distributor runs simultaneously to have the reliable ZMP value.

Through three control phases, the robust posture of the humanoid is achieved on even or rough surface. Because of the robust posture, the performance of motion and balance controller in manipulation tasks can be guaranteed.

6.6   CRPC: Experimental Verification and Validation

To verify and validate the control approach proposed in this paper, we emulated the rough terrain, sloped terrain, and uneven surface conditions previously described. The Hubo was initially calibrated and standing on such surfaces as shown in A, C, and E in Figure 6.11, respectively. To emulate rough terrain, a half-inch thick aluminum plate was placed at the toe of right foot. In the posture that results from standing on this rough terrain, the arbitrary moments are initially acting on the right and left feet and the weight of body is unequally distributed. In this case, the performance of any balance controller cannot be guaranteed and external disturbances from reaction forces that result during manipulation tasks can result in the instability of the humanoid. Thus, such a posture has to be adjusted before performing a manipulation task. To achieve the robust posture conditions defined in Section 6.4, the Cascade Robust Posture Controller (CRPC) is applied to the posture on rough surface (Figure 6.11A). In this case, for the CRPC to produce a stable posture, the desired upper-body orientation is upright. Figure 6.11B depicts the robust posture
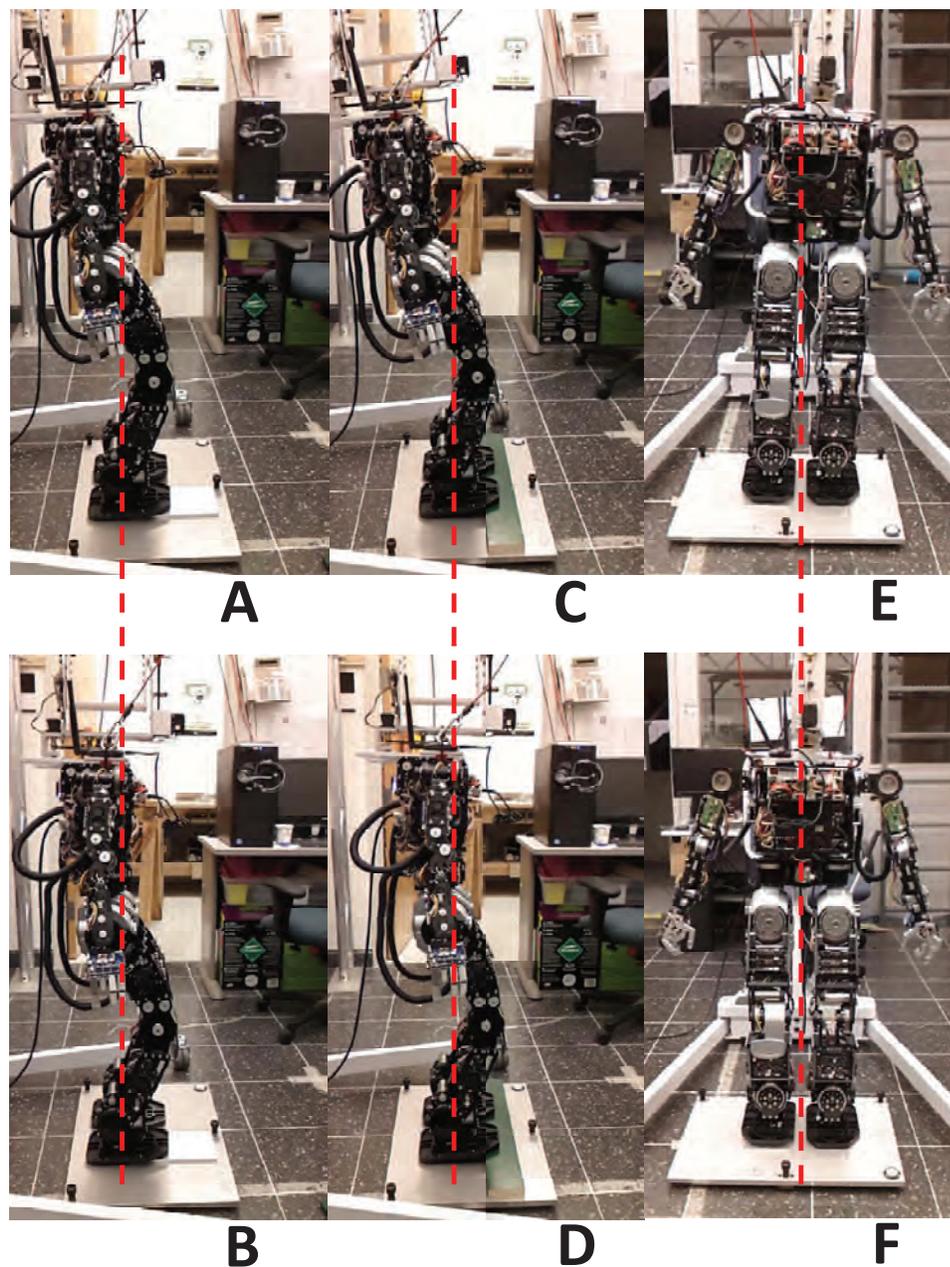
Figure 6.11: Three surface conditions with a full-sized humanoid, Hubo2. The dotted red line is the center of the support polygon: A) Initial posture on rough terrain. B) Robust posture on rough terrain achieved by Cascade Robust Posture Control (CRPC). C) Initial posture on inclination. D) Robust posture achieved by CRPC on inclination. E) Initial posture on uneven terrain. F) Robust posture obtained using (CRPC) for uneven terrain.
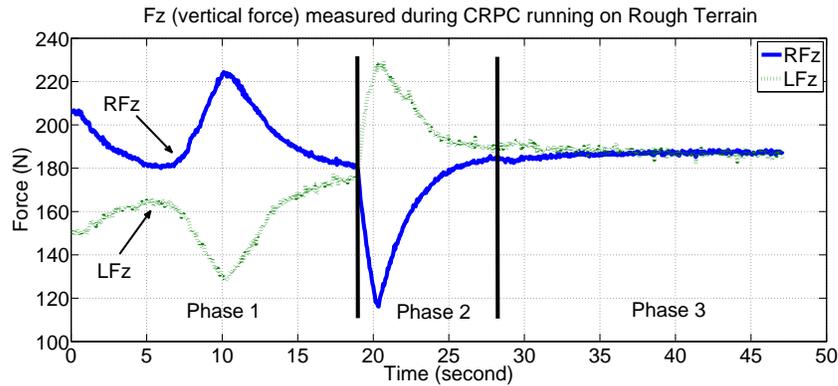
Figure 6.12: The vertical force (F_z) measured from F/T sensors on the right, RF_z, and left, LF_z, feet during CRPC running for rough surface condition.

Figure 6.13: Moment plots exerted on the right, RMy and RMx, and left, LMy and LMx, feet in CRPC for rough surface. My and Mx denote a moment around Y (lateral) and X (forward) axes, respectively.

Figure 6.14: Zero-Moment Point graph in Double-Support Phase during CRPC running on rough surface. ZMP_x and ZMP_y are the ZMP in X (forward) and Y (lateral) direction.

on rough terrain. The sensor measurements of the reaction force (F_z) and moments (My and Mx) on the feet, and the estimated ZMP data are shown in Figure 6.12, Figure 6.13, and Figure 6.14, respectively. These graphs demonstrate that the posture in Figure 6.11B is a robust posture. The weight is equally distributed (Figure 6.12), all moments exerted on both feet are minimized to zero (Figure 6.13), and the ZMP is converges to zero (Figure 6.14). From the robust posture, the actual ZMP and CoP are coincide as well, and the maximum robustness of the humanoid against dynamic influences is achieved.

Figure 6.11C is the initial posture of the humanoid on the inclined surface. A one inch thick wood bar is located randomly under the toe of both feet. As a result, the slope forces the humanoid to lean backward. After running the CRPC, the humanoid stands up robustly (Figure 6.11D). Figure 6.15, Figure 6.16, and Figure 6.17 are the sensor measurements and ZMP data associated with the postures on the inclination while the CRPC is applied. This data also confirms that the posture in Figure 6.11 D satisfies the robust posture conditions defined in Section 6.4.

Figure 6.11E and F show the initial and final postures on uneven surfaces, respectively. A half-inch thick aluminum plate is placed under the right foot and it results in the humanoid leaning in lateral direction. After the CRPC is applied, the posture in Figure 6.11E is adjusted as Figure 6.11F which satisfies the robust posture conditions. The sensor data corresponding to this experiment is depicted in Figure 6.18, Figure 6.19, and Figure 6.20.

Note that the tolerance is $\pm 2N$ for the vertical forces, F_z, and $\pm 0.2N \cdot m$ for moments, M_y and M_x, on each foot. For the ZMP and upper body orientation, $\pm 2mm$ and $\pm 0.1\deg$ are also set as the tolerance. To minimize the dynamic coupling in the CRPC, all gains for each independent controller in Section 6.5 are set as smaller than 10^{-3} in the experiments.
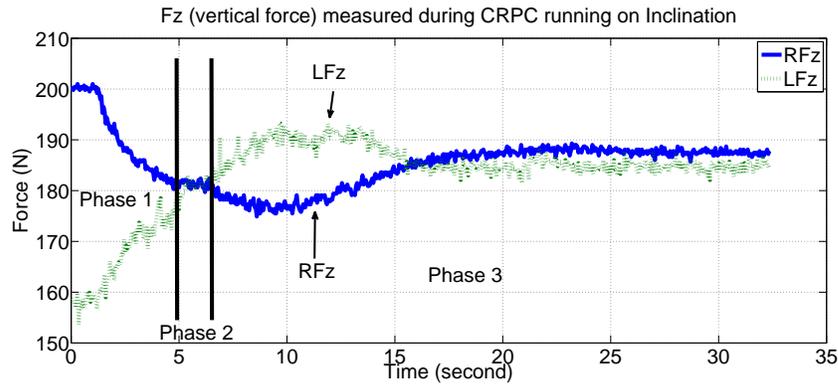
Figure 6.15: The vertical force (F_z) measured from F/T sensors on right (blue-bold) and left (green-dot) feet during CRPC running for inclination.
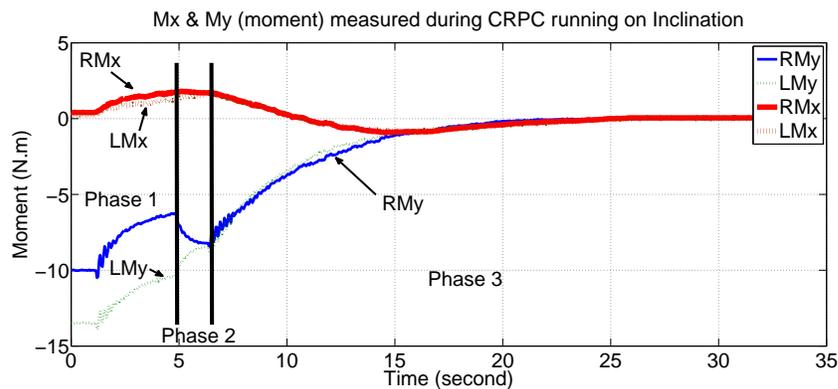
Figure 6.16: Moment plots exerted on right, RMy and RMx, and left, LMy and LMx, feet in CRPC for slope surface. My and Mx denote a moment around Y (lateral) and X (forward) axes, respectively.
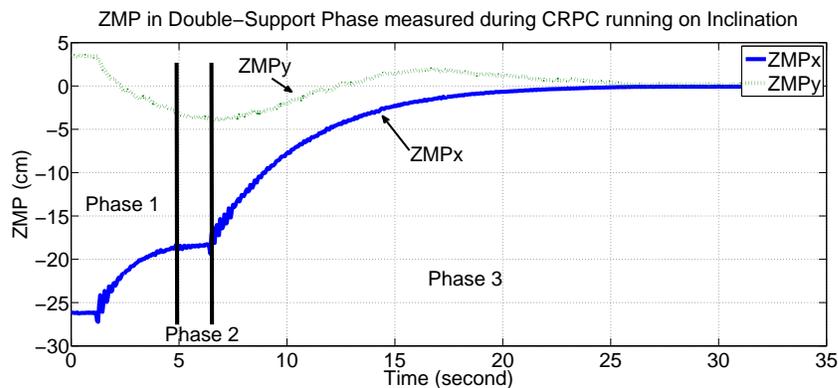
Figure 6.17: Zero-Moment Point graph in Double-Support Phase during CRPC running on slope. ZMP_x and ZMP_y are the ZMP in X (forward) and Y (lateral) direction.
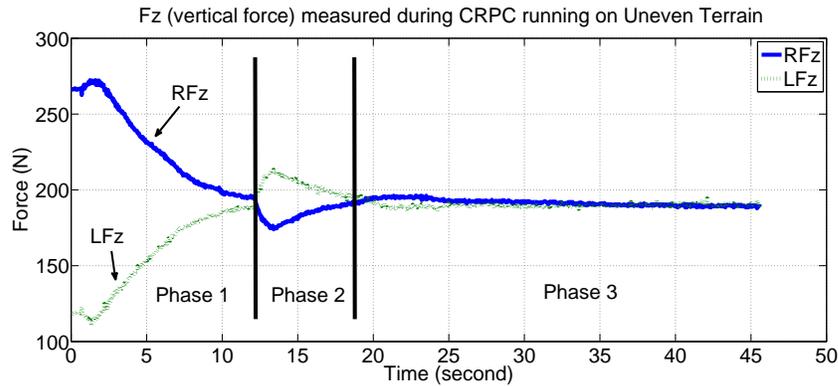
Figure 6.18: The vertical force (F_z) measured from F/T sensors on right (blue-bold) and left (green-dot) feet during CRPC running in uneven condition.
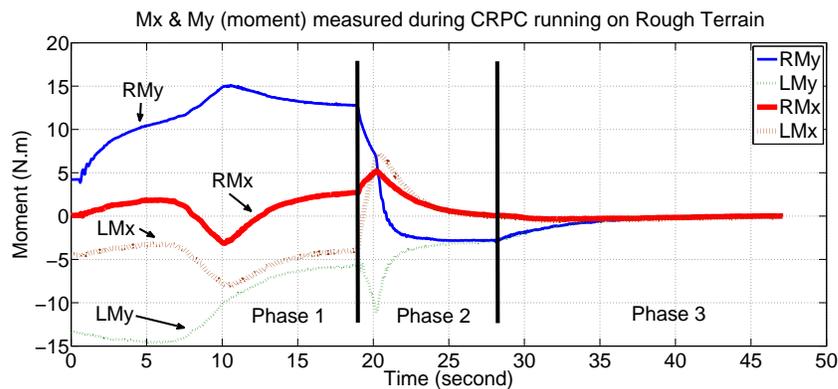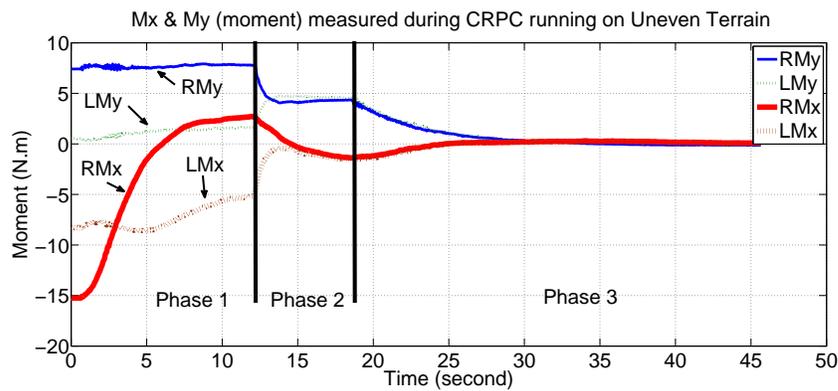

Figure 6.19: Moment plots exerted on right, RMy and RMx, and left, LMy and LMx, feet in CRPC for uneven surface. My and Mx denote a moment around Y (lateral) and X (forward) axes, respectively.
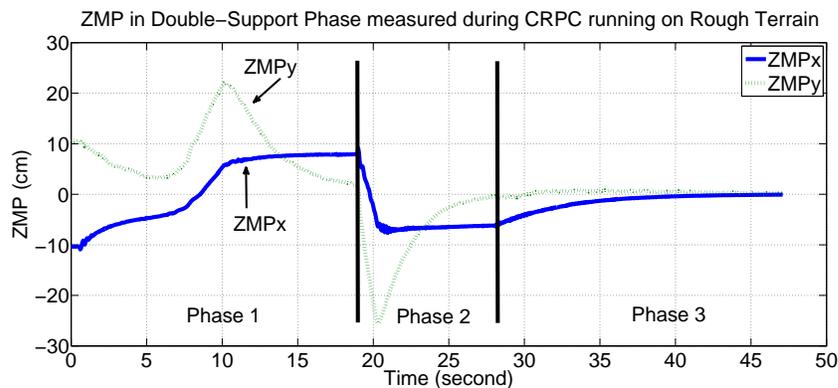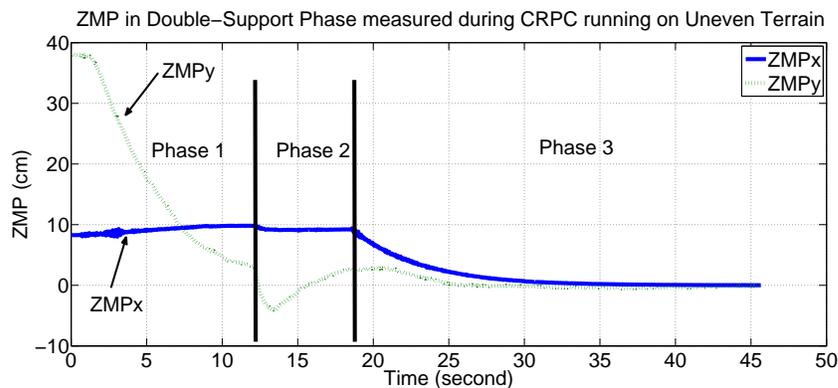

Figure 6.20: Zero-Moment Point graph in Double-Support Phase during CRPC running on uneven surface. ZMP_x and ZMP_y are the ZMP in X (forward) and Y (lateral) direction.
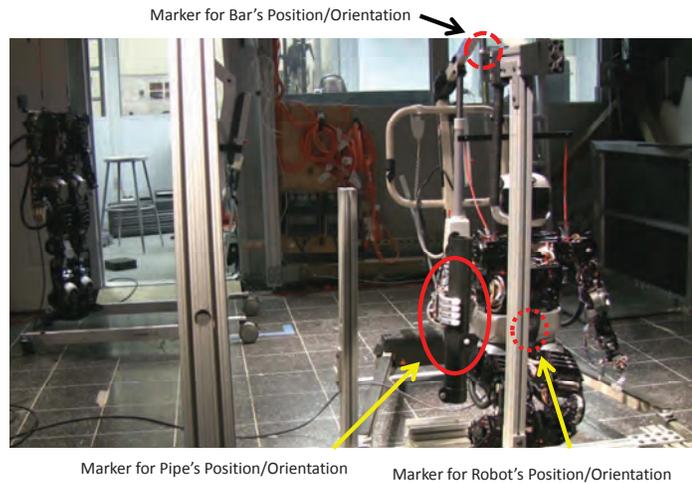
Figure 6.21: Whole-body peg-in-hole task setup under motion capture system.

6.7 Peg-In-Hole Task under Motion Capture System

The peg-in-hole task is a well-known problem often used to quickly test and evaluate one's ideas and approaches. The peg-in-hole task for humanoids is demonstrated here as a fist step in the research toward humanoids autonomously performing a task in human-centered environments. The problems and technical approaches are rapidly prototyped in simulation, tested and evaluated in a full-sized humanoid, and verified and validated with Motion Capture (MoCap) system.

Peg-in-hole task is simply to insert a pipe into a hole. It requires perception of surroundings like positions and orientations of the robot and target, path planning, and motion planning and control. Figure 6.21 shows the experimental setup. A humanoid, pipe, and bar are located under MoCap system. The red-circles indicate the location of markers for them. MoCap system detects these markers and returns marker's positions and orientations in global frame to the controller. Constrained Bi-directional Rapidly-exploring Random Trees (CBiRRT) in OpenRave simulator
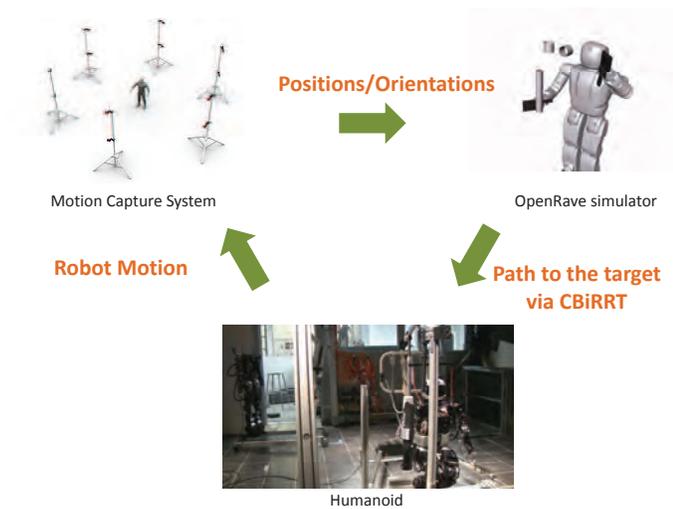
Figure 6.22: Layout of system operation for autonomous peg-in-hole task

plans a path from the pipe to bar according to the position and orientation data. The trajectory planned is sent to the robot to follow it. A balance controller stabilizes the robot in real-time. The controller adjusts the CoM position while the robot follows the given trajectory. This results in a misalignment between the pipe and bar at the end of trajectory following. MoCap system sends the position and orientation data to the robot again and the pelvis position is adjusted to compensate the error. Figure 6.22 shows the layout of system operation. Figure 6.23 is a snapshot of Hubo autonomously performing this task.
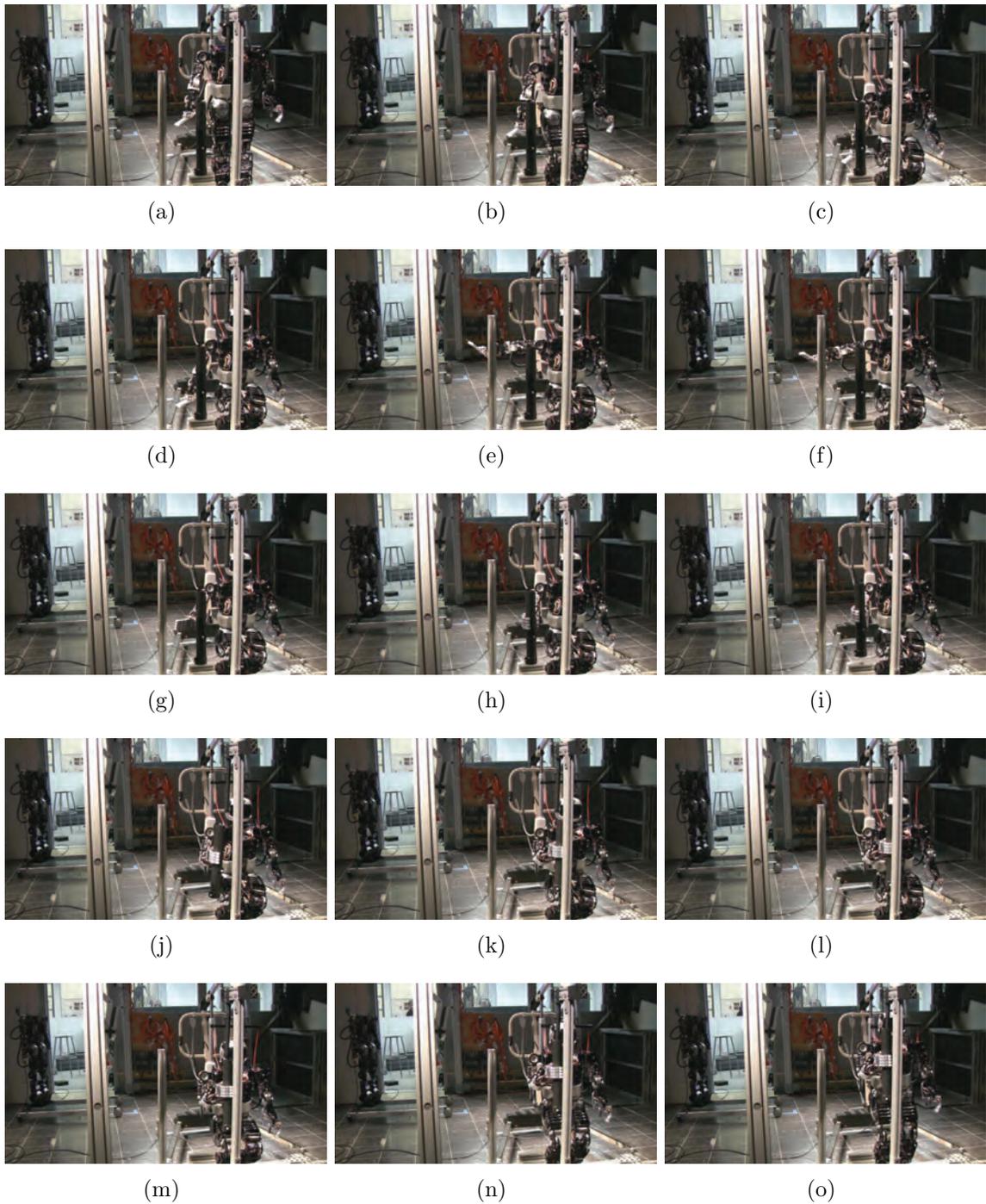
Figure 6.23: Autonomous peg-in-hole task under Motion Capture system.

7. TELE-OPERATED HUMANOID IN REAL-WORLD APPLICATIONS

Humanoids are being developed with ever increasing capabilities. Current humanoids are capable of sensing and interacting at a very high level with increasingly complex environments. While the long term goal is to make these robots fully autonomous, it remains very challenging to implement this. Teleoperation, with shared autonomy with a human controller, is an emerging paradigm that has proved useful in allowing humans to control robots performing complex tasks.

This chapter describes a teleoperated interface that combines high level sensing and low level control. This semi-autonomous system allows a humanoid to perform a task in human-centered environments. This abilities are demonstrated through teleoperated hose installation task and biped locomotion in outdoor environments. These approaches were used for DARPA Robotic Challenges 2013.

7.1 Tele-Operational System Architecture

Tele-operational system consists of three different individuals, each managing their own tool chain. The layout of the team and the strategy is shown in Figure 7.1. The task is broken down to piloting, sensor management, and situational awareness. The pilot is in charge of the task completion and control over the main actuators of the robot. The sensor manager controls the head of the robot to gather vision sensor information. The situational awareness manager organizes the information gathered by the vision sensors so that it can be used by the pilot.
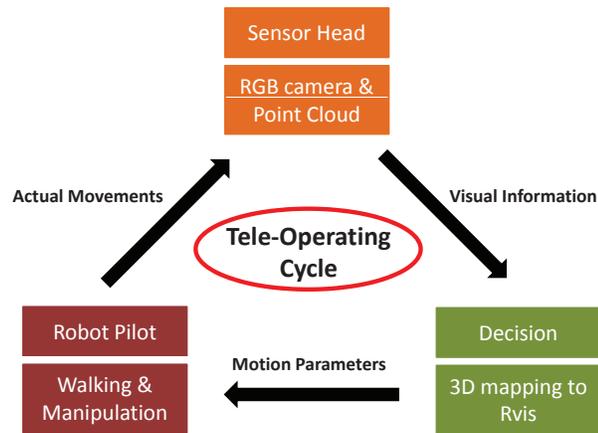
Figure 7.1: Layout of system operation. One person is responsible for generating sensor data and visual measurement, a second person organizes the information to generate situational awareness, and a third person who is responsible for generating actual motion commands.

Sensor Head Manager

The sensor head is described in Appendix A. It is made up of a pan-tilt neck with an integrated IMU, and an additional motor for a tilting laser scanner. The head carries 3 stereo cameras along with a Hokuyo laser scanner. The outdoor stereo cameras allow an operator to visually perceive surroundings under shadows or direct sunlight. The laser scanner scans environments. It returns a set of data points (point cloud) in robot-referenced frame. Such data points map the three-dimensional environments and numerically represent it.

The sensor head manager is tasked with monitoring the quality of the communication channel to the robot and varying the sensor feedback's bandwidth usage accordingly to maintain the system's overall responsiveness. The sensor payload of the head is expensive and fragile, so while the robot is walking long distance or in
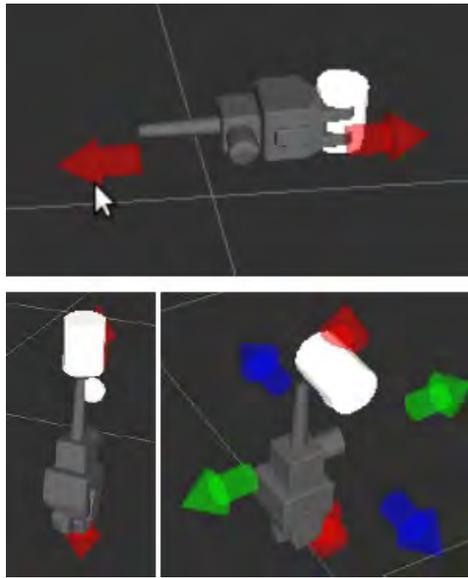
Figure 7.2: An illustration of the 3 manipulation fixtures associated with a cylinder object. On top is a power grasp of the hose end and positioning the hose end the environment. Bottom left is a hooking motion for not prehensile grasping by inserting the peg in the end of the hose. On the bottom right is a starting posture for pushing the rotating threads at the end of the hose.

uncertain terrain, the sensor head's actuators are placed into a compliant mode, so impact of environmental collisions or falls is minimized. The sensor head manager controls the actuators on the head to point the sensors as directed by the pilot. The interface for the sensor head control is a set of custom dialogs implemented in RViz [93], where the pilot and sensor head manager visualize the current scene, as well as feedback provided by the situational awareness manager.

Situational Awareness Manager

In order to allow the pilot to make accurate decisions, the situational awareness manager aligns primitives to the point cloud provided by the sensor head. Multiple point clouds can be accumulated and manually aligned to one another. After roughly aligning entities, ICP can be used to refine this alignment. This manager uses 'virtual
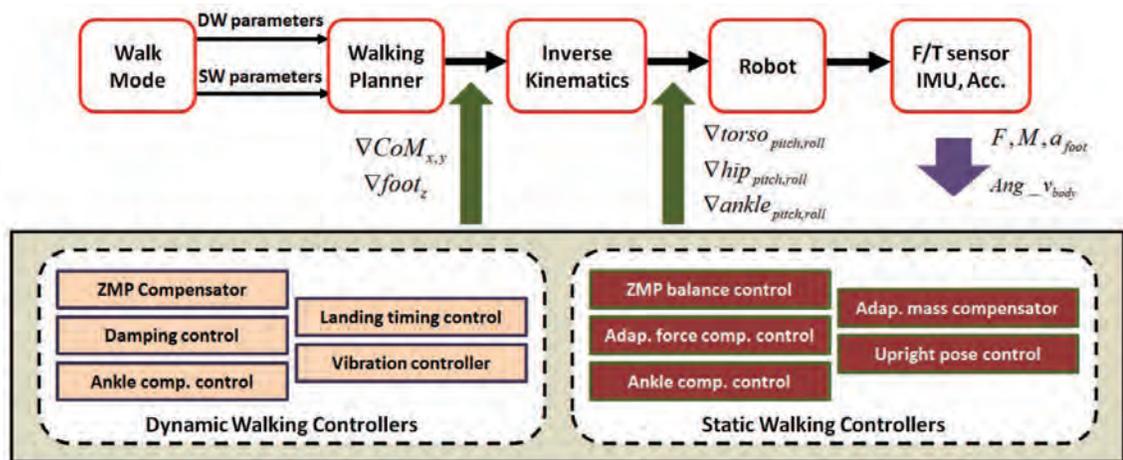
Figure 7.3: Block diagram of walking control mode: dynamic and static walking.

fixture' approach [94]. A virtual fixture is analogous to using a ruler draw a straight line on a piece of paper. It is an element of an interface in a virtual reality environment that constrains the operators ability to manipulate objects in the scene to enforce constraints like alignment and reachability. The motion planner is invoked to test reachability to the target poses viable for the manipulation type directed by the pilot, and if a reasonable, reachable motion plan can be found, it is reviewed by the pilot for execution. For fine motions, the pilot can visualize the target pose and the Cartesian distance to the current end effector pose for input to the online inverse kinematics controller. If there is no reachable plan, the situational awareness manager has tools to analyze the cause of the failure and suggest corrective actions to the pilot. Figure 7.2 illustrates the manipulation strategy using virtual fixture in simulator, performed by situational awareness manager.

Piloting

The pilot has controls the robot base's motion through two Graphic User Interfaces (GUI) for biped locomotion and manipulation. In biped locomotion, the dynamic

and static walking modes are designed. Figure 7.3 describes the control schematic for biped locomotion containing real-time feedback controllers. The dynamic mode is used to quickly maneuver on flat surface while the static mode is more stable and adaptable for locomoting on rough terrains. Based on visual information, such modes are switched by an operator. From situational awareness manager, gait parameters like stride, turning angle, slope, and ground height are commanded. In manipulation, Operational Space Formulation (OSF) is implemented to control a high DOF redundant joints manipulator. It controls whole body to follow a given trajectory or end-effector position and orientation command from situational awareness manager. The motion trajectory is planned using CBiRRT with collision checking in OpenRave simulator.

7.2 Tele-Operational Hose Installation Task

Hose installation task is a task that requires humanoid's capabilities to locomote, manipulate, and carry an object. Performing this task perhaps demonstrates how the tele-operated humanoid developed on methods in this dissertation fits to work in human-centered environments. This task is broken down into several sub-tasks according to the objectives of motion: approach, acquisition, transportation, and connection. The field of task environment is depicted in Figure 7.4 provided by Defense Advanced Research Projects Agency (DARPA)[1]. The robot initially stands behind the green line. The robot approaches to the real attached on the wall, where the hoes is hanging down. After acquiring the hose, the robot drags it to Y shaped

---

[1]DARPA Robotic Challenge (DRC) in 2013 is the world-wide biggest robotic challenge. It was a milestone for robotists to put an effort on robotic systems disaster response. There were 8 missions under a disaster scenario; driving, rough terrain, debris removal, door opening, ladder climbing, wall breaking,valve turning, and hose installation. The author led two missions; rough terrain and hose installation and a tele-operational system was implemented through collaboration with Columbia University, Ohio State University, and University of Delaware. The author served as a main mechanic for managing the robot, and developer/operator of locomotion for valve turning (WPI) and ladder climbing (Purdue University) [17].
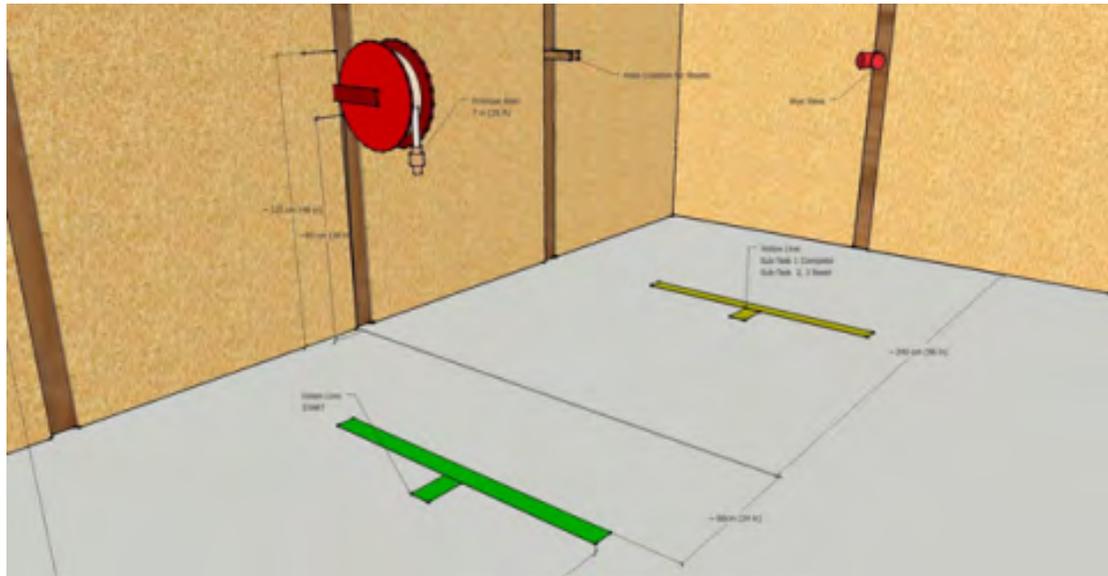
Figure 7.4: A diagram of the hose task from the DRC task description. The robot begins behind the green line, walks forward to grasp the hose hanging on the reel, then walks across the yellow line to attach the hose to the wall mounted Y shaped water outlet, called a wye connector

water outlet, called a wye connector. In last, the robot connects the hose end to the wye. The shape and dimension of objects and tools like reel, hose, and wye are pre-specified. But, the locations of them and ground condition are arbitrary.

Approach to the Reel

A operator visually recognizes the environments via stereo cameras. The location of the target and ground condition are checked. The operator decides a walking mode and commands gait parameters like steps, stride, and turning angle to the robot. The robot produces a stable gait pattern according to the given commands. Real-time feedback controllers make the robot fully stable and adaptive to the feature of environments.

Acquisition

The virtual fixtures allows the scene manager to quickly and accurately show the pilot reasonable goal poses for the manipulator. The scene manager accomplishes this by aligning simplified representations of the objects to manipulated with the point cloud provided by the sensor head manager. The goal position and orientation of the end-effector (hand) are determined from this alignment in 3D simulator. A path planner or manual operation allows the end-effector to move to the goal pose. Then, the robot can grasp the hose from the reel. This manipulation is a pure kinematic motion.

Transportation

It is a phase to drag a hose. The rotational resistance of the reel is very small but the friction between the hose and ground generates an arbitrary force in opposite to the direction the robot walks toward. The CoM position of the robot is moved to the opposite direction to the force direction in order to balance against such external force.

Connection

It is a motion phase to align hose end to the rotational connector (wye). This manipulation requires extremely accurate. A trajectory is planned to move the hose end close to the wye. Then, the manipulation control mode is switched to position mode to force mode. Once the hose end touches the wye with a certain force, the other end-effector rotates the coupling based on a given trajectory planned.

Figure 7.5, Figure 7.7, Figure 7.8 are a snapshot of laboratory verification and validation, dry-run, and DRC trial in 2013. Figure 7.6 depicts the robot and environments mapped into the simulator. The goal pose is determined in the simulator

(a)	(b)	(c)
(d)	(e)	(f)

Figure 7.5: Tele-operated DRC-Hubo Hose installation task experiment.

using virtual fixture approach.

7.3 Tele-Operational Biped Locomotion on Rough Terrain

Rough terrain task here consists of ascent/descent ramp, obstacles stepped over, and ascent/descent stairs. This task requires a stable, reliable, and adaptable biped locomotion that perhaps demonstrates humanoid's ability to maneuver in human-centered environment. Figure 7.9 depicts the field of terrains. The robot starts from the ramp and ends with descent stairs in this experiment. The humanoid is fully tele-operated. Sensor head manager visualizes the terrains with the stereo cameras and scans the ground roughness with a laser sensor. Situational awareness manager uses virtual fixture to approximately estimate the dimension, distance, and slope of terrains in Cartesian space. Pilot chooses the walking mode according to terrain information and commands gait parameters such as number of steps, strides, angles to turn, and slope to the humanoid. The humanoid plans the biped motion and real-time feedback controllers stabilize it and compliance the force acting on the foot.
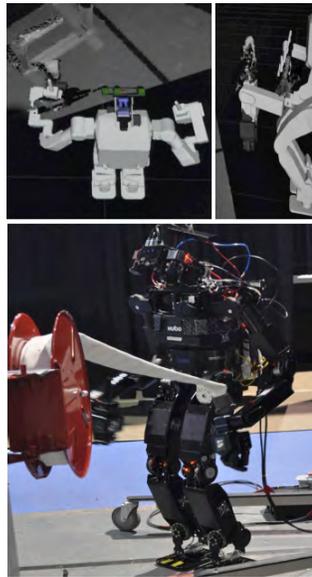
Figure 7.6: (Bottom) DRC-Hubo performing the hose task in dry-run. (top left) feedback from the vision system during hose acquisition. (top right) the view of the robots left hand spike aligned with the hose end during hose acquisition.

| (a) | (b) | (c) |
| (d) | (e) | (f) |
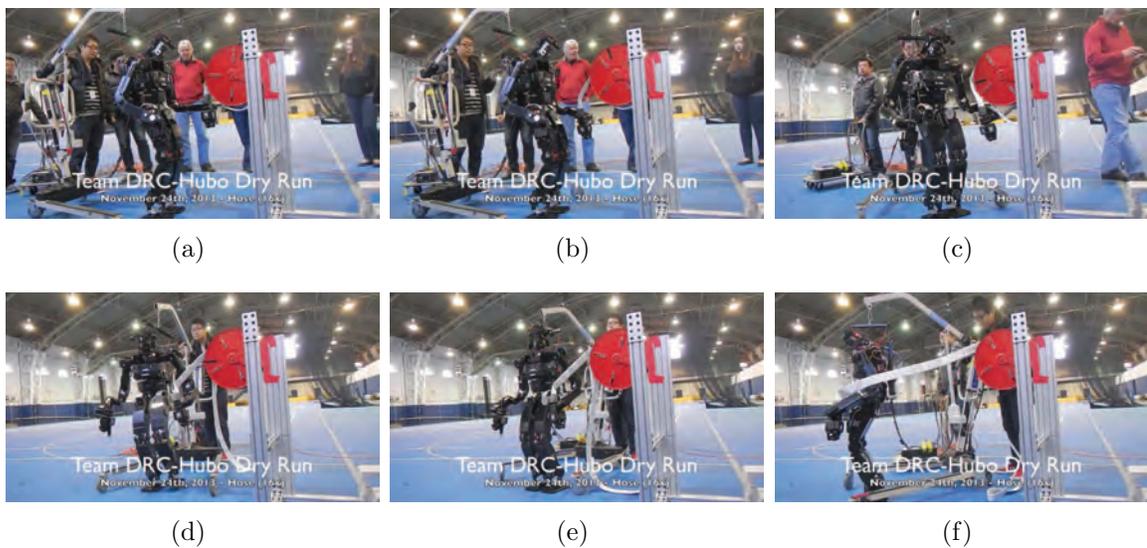
Figure 7.7: DRC-Hubo, Hose installation task in dry-run.

(a)

(b)

(c)

(d)
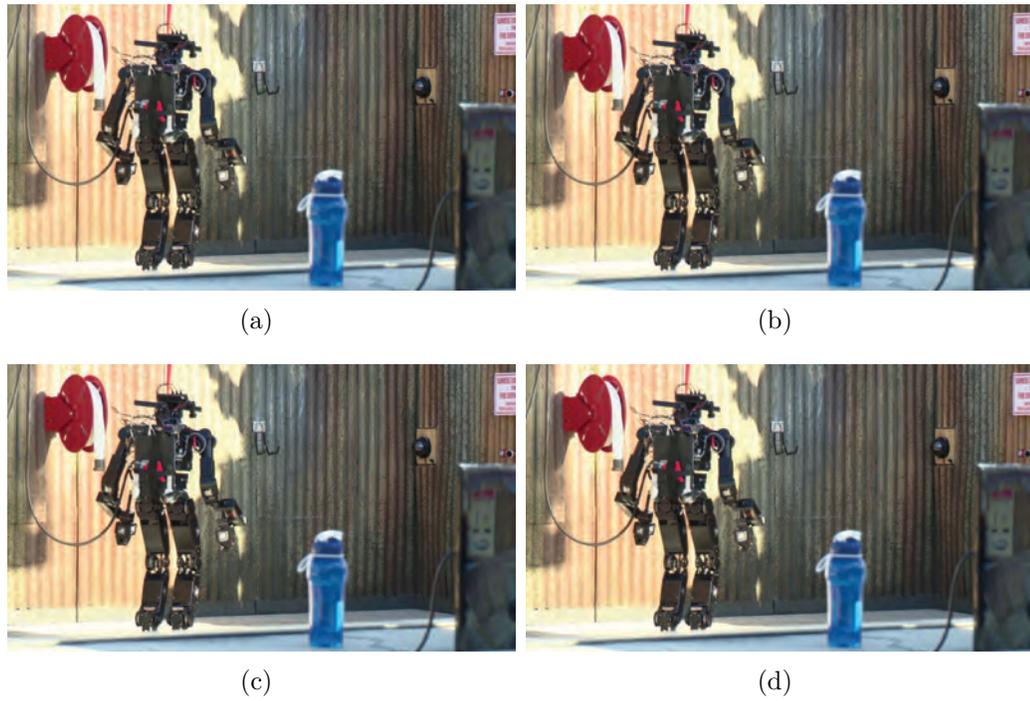
Figure 7.8: DRC-Hubo performing a hose installation task in DRC trial 2013.
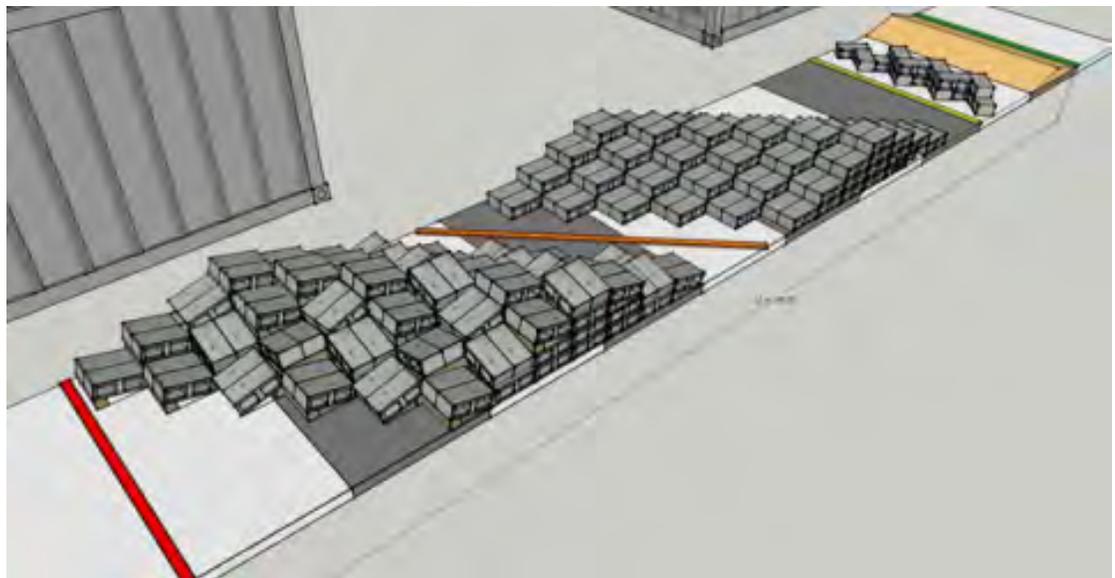
Figure 7.9: The field of rough walking task provided DARPA Robotic Challenge 2013.

Figure 7.10, Figure 7.11, Figure 7.12 are a snapshot of evaluation of the biped locomotion performance, dry-run, and DRC trial in 2013.
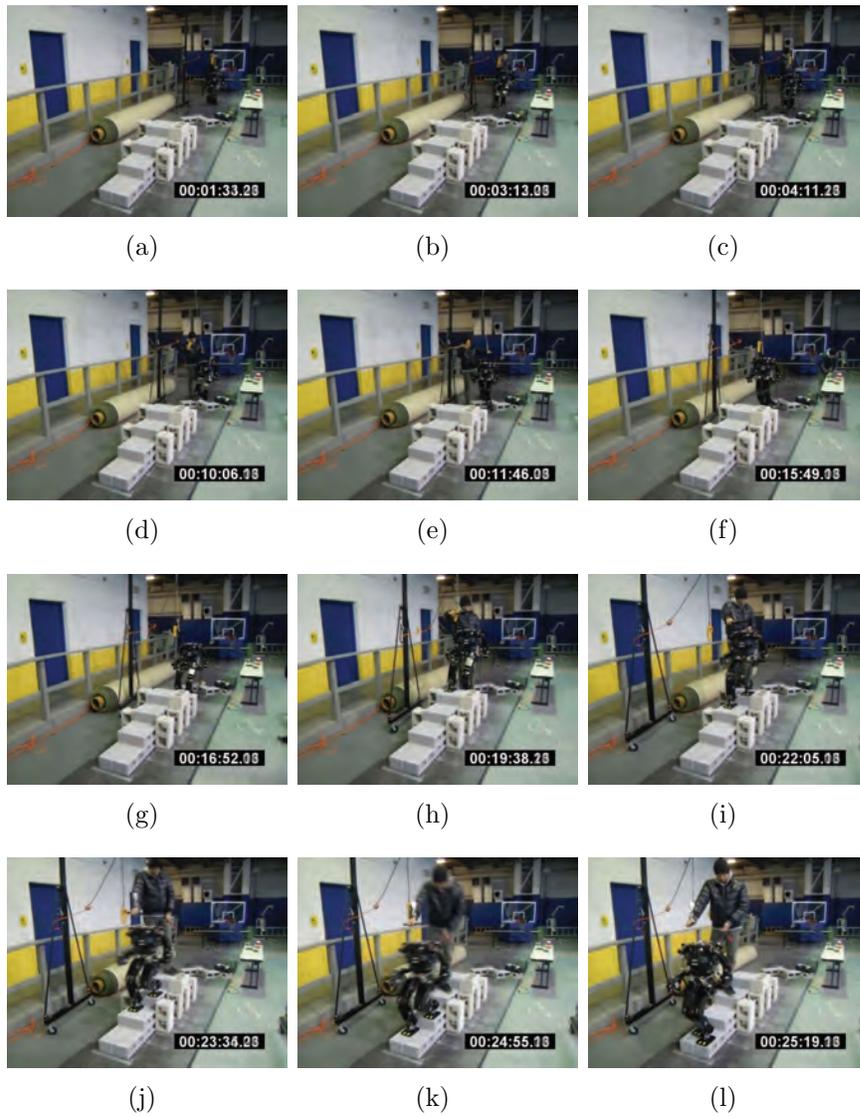
Figure 7.10: Performance evaluation of biped locomotion on rough terrains.

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)
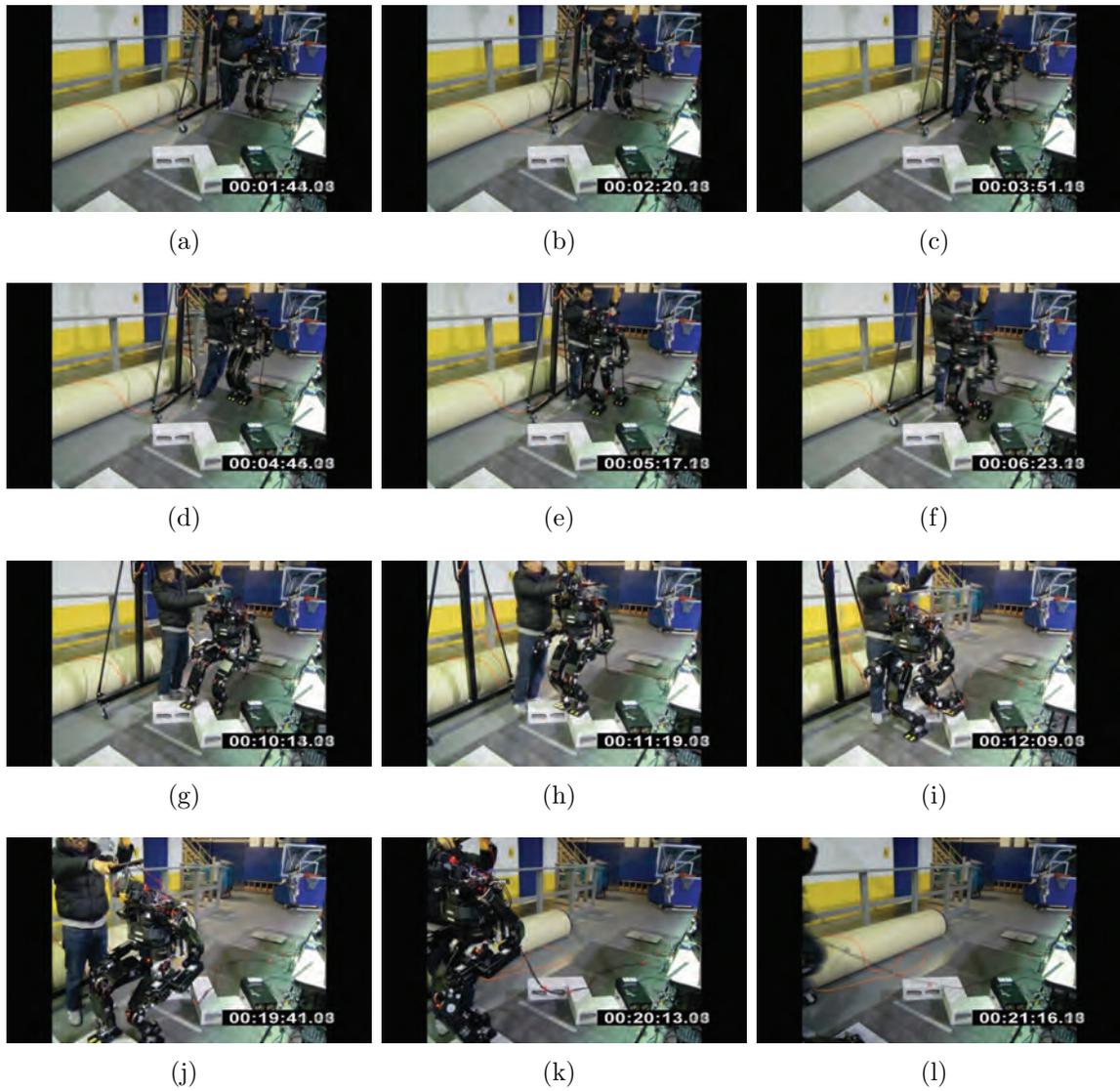
Figure 7.11: Tele-operated DRC-Hubo walking on rough terrain.
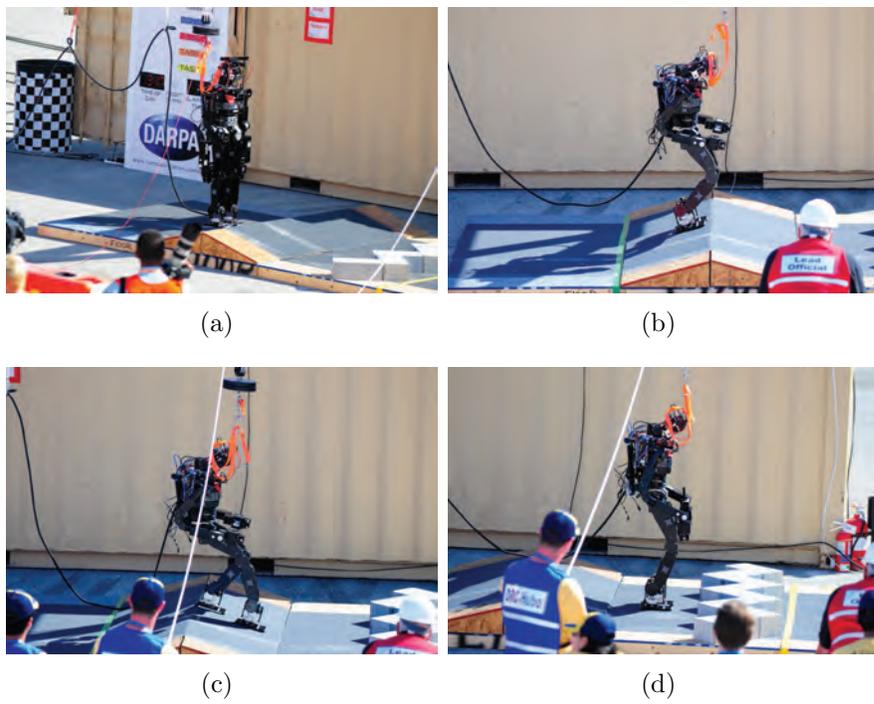
Figure 7.12: DRC-Hubo walking on ramp in DRC Trial 2013.

8. CONCLUSION

Conclusion

This dissertation demonstrated technical approaches toward humanoids working in human-centered environments. The dissertation proposed that the practical use of humanoids requires an integration of perceptibility, maneuverability, and manipulability which must be reliable, stable, and adaptable to arbitrary environments. The model-based gait planning algorithm planned a stable biped locomotion that allowed a humanoid to maneuver within indoor and outdoor environments. Real-time feedback controllers stabilized the robot against external forces generated by environmental interactions, minimized the mechanical disturbances like vibration, and allowed for compliance of impulsive forces during the moment of contact with objects. Maneuverability was realized through integration of the aforementioned techniques and was fully tested-and-evaluated, and verified-and-validated through both dynamic and static locomoting while ascending and descending stairs and ramps and stepping over obstacles. With these controllers, the motion of whole-body manipulation was realized. Posture controllers allowed the robot to adjust the magnitude of force and have a robust posture against disturbances on uncertain terrains. Based on a path planning algorithm and IK-based whole-body motion control, a humanoid was able to push a heavy object and autonomously perform a peg-in-hole task with feedback from a motion capture system. This experimental demonstration showed humanoid's capability to manipulate objects and tools using supplied geometric information. With these low-level reliability, stability, and adaptability in whole-body motion, a teleoperational system for high-level commands was implemented to compensate for the lack of artificial intelligence. A sensor head manager visualized the surroundings, a

situational awareness manager estimated the geometric data like the positions and orientations of the robot and target, and pilot decided the control modes and commanded the robot to move.

This semi-autonomous humanoid successfully completed a complex manipulation task (hose installation) and locomotion task (rough/obstacle terrain walking). In conclusion, this dissertation clearly pointed out the critical concepts and requirements toward humanoids autonomous performing tasks in practical applications. The technical design approaches theoretically developed and experimentally verified-and-validated in this dissertation successfully satisfied the requirements and filled the gaps in the current generation of state-of-the-art humanoids. This dissertation was a step toward humanoids autonomously assisting, interacting, and living in human-centered environments.

Bibliography

[1] N. Alunni, C. Phillops-Graffin, H. Bener, D. Lofaro, D. Berenson, S. Chernova, R. Lindeman, and P. Oh, "Toward a user-guided manipulation framework for high-dof robots with limited communication," in *Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference on*, Apr 2013, Conference, pp. 1–6.

[2] H. Arisumi, J. Chardonnet, A. Kheddar, and K. Yokoi, "Dynamic lifting motion of humanoid robots," in *Robotics and Automation, 2007. Proceedings. 2007 IEEE International Conference on*, Apr 2007, Conference, pp. 2661–2667.

[3] H. Arisumi, S. Miossec, J. Chardonnet, and K. Yokoi, "Dynamic lifting by whole body motion of humanoid robots," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, Sep 2008, Conference, pp. 668–675.

[4] ——, "Whole-body motion of a humanoid robot for passing through a door: opening a door by impulsive force," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, Oct 2009, Conference, pp. 428–434.

[5] T. Asfour and R. Dillmann, "Human-like motion of a humanoid robot arm based on a closed-form solution of the inverse kinematics problem," in *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 2, Oct 2003, Conference, pp. 1407–1412.

[6] D. Berenson, J. Chestnutt, S. Srinivasa, J. Kuffner, and S. Kagami, "Pose-constrained whole-body planning using task space region chains," in *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*, Dec 2009, Conference, pp. 181–187.

[7] D. Berenson, S. Srinivasa, D. Ferguson, and J. Kuffner, "Manipulation planning on constraint manifolds," in *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, May 2009, Conference, pp. 625–632.

[8] BostonDynamics. [Online]. Available: http://www.bostondynamics.com/robot$\_$Atlas.html

[9] S. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," *IEEE Journal of Robotics and Automation*, 2004.

[10] Y. Cha, S. Hong, D. Kim, B. You, and S. Oh, "Whole-body humanoid balance control with dynamically loading/unloading objects," in *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, Dec 2009, Conference, pp. 1687–1691.

[11] Y. Chang, Y. Oh, D. Kim, and S. Hong, "Balance control in whole body coordination framework for biped humanoid robot mahru-r," in *Robot and Human Interactive Communication, 2008. RO-MAN 2008. The 17th IEEE International Symposium on*, Aug 2008, Conference, pp. 401–406.

[12] B. Cho, S. Park, and J. Oh, "Controllers for running in the humanoid robot, hubo," in *Humanoid Robots 2009, 9th IEEE-RAS International Conference on*, Dec 2009, Conference, pp. 385–390.

[13] Y. Choi, D. Kim, and B. You, "On the walking control for humanoid robot based on the kinematic resolution of com jacobian with embedded motion," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, May 2006, Conference, pp. 2655–2660.

[14] Y. Choi, B. You, and S. Oh, "On the stability of indirect zmp controller for biped robot systems," in *Intelligent Robots and Systems, 2004, Proceedings. 2004 IEEE International Conference on*, vol. 2, June 2004, Conference, pp. 3–10.

[15] H. Dallali, M. Brown, and B. Vanderborght, "Using the torso to compensate for non-minimum phase behaviour in zmp bipedal walking," *Journal of Advancs in Robotics Research*, vol. 2, pp. 191–202, 2009.

[16] H. Dang, Y. Jun, P. Oh, and P. Allen, "Planning complex physical tasks for disaster response with a humanoid robot," in *Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference on*, Apr 2013, Conference, pp. 1–6.

[17] DARPA. [Online]. Available: http://www.theroboticschallenge.org/

[18] K. Erbatur and O. Kurt, "Natural zmp trajectories for biped robot reference generation," *Industrial Electronics, IEEE Transaction on*, vol. 56, no. 3, pp. 835–845, Mar 2009.

[19] P. Evrard, E. Gribovskaya, S. Calinon, A. Billard, and A. Kheddar, "Teaching physical collaborative tasks: object-lifting case study with a humanoid," in *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*, Dec 2009, Conference, pp. 399–404.

[20] C. Fu and K. Chen, "Gait synthesis and sensory control of stair climbing for a humanoid robot," *Industrial Electronics, IEEE Transaction on*, vol. 55, pp. 2111–2120, May 2008.

[21] A. Goswami and V. Kallem, "Rate of change of angular momentum and balance maintenance of biped robots," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 4, Apr 2004, Conference, pp. 3785–3790.

[22] Y. Gu, T. Anand, and W. Sheng, "Human-robot collaborative manipulation through imitation and reinforcement learning," in *Information and Automation (ICIA), 2011 IEEE International Conference on*, Jun 2011, Conference, pp. 151–156.

[23] Y. Guan, E. Neo, K. Yokoi, and K. Tanie, "Stepping over obstacles with humanoid robots," *Robotics, IEEE Transactions on*, vol. 22, no. 5, pp. 958–973, Oct 2006.

[24] K. Harada, S. Kajita, F. Kanehiro, K. Fujiwara, K. Yokoi, and H. Hirukawa, "Real-time planning of humanoid robot's gait for force-controlled manipulation," *Mechatronics, IEEE/ASME Transaction on*, vol. 12, no. 1, pp. 53–62, Feb 2007.

[25] K. Harada, S. Kajita, K. Kaneko, , and K. Hirukawa, "Pushing manipulation by humanoid considering two-kinds of zmps," in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 2, Sep 2003, Conference, pp. 1627–1632.

[26] K. Harada, S. Kajita, H. Saito, M. Morisawa, F. Kanehiro, K. Fujiwara, K. Kaneko, and H. Hirukawa, "A humanoid robot carrying a heavy object," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, Apr 2005, Conference, pp. 1712–1717.

[27] K. Harada and M. Kaneko, "Whole body manipulation," in *Robotics, Intelligent Systems and Signal Processing, 2003. Proceedings. 2003 IEEE International Conference on*, vol. 1, Oct 2003, Conference, pp. 190–195.

[28] K. Hashimoto, Y. Sugahara, A. Ohta, H. Sunazuka, C. Tanaka, M. Kawase, L. H.O., and A. Takanishi, "Realization of stable biped walking on public road with new biped foot system adaptable to uneven terrain," in *Biomedical Robotics and Biomechatronics, 2006. BioRob 2006. The First IEEE/RAS-EMBS International Conference on*, Feb 2006, Conference, pp. 226–231.

[29] K. Hashimoto, Y. Sugahara, H. Sunazuka, C. Tanaka, A. Ohta, M. Kawase, H. Lim, and A. Takanishi, "Biped landing pattern modification method with nonlinear compliance control," in *Robotics and Automation, 2006. ICRA 2006.*

Proceedings 2006 IEEE International Conference on, May 2006, Conference, pp. 1213–1218.

[30] K. Hashimoto, Y. Takezaki, H. Motohashi, T. Otani, Y. Kishi, H. Lim, and A. Takanishi, "Biped walking stabilization based on gait analysis," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, May 2012, Conference, pp. 154–159.

[31] J. Hing and P. Oh, "Mixed reality for unmanned aerial vehicle operations in near earth environments," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, Oct 2010, Conference, pp. 2537–2538.

[32] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka, "The development of honda humanoid robot," in *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, vol. 2, May 1998, Conference, pp. 1321–1326.

[33] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, "Quadrotor helicopter flight dynamics and control: Theory and experiment," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, vol. 2, 2007, Conference.

[34] N. Hogan, "Impedance control: An approach to manipulation," *American Control Conference, 1984*, pp. 304–313, 1984.

[35] Q. Huang, S. Kajita, N. Koyachi, K. Kaneko, K. Yokoi, H. Arai, K. Komoriya, and K. Tanie, "A high stability, smooth walking pattern for a biped robot," in *Robotics and Automation, 1999, Proceedings. 1999 IEEE International Conference on*, vol. 1, 1999, conference, pp. 65–71.

[36] Q. Huang, Y. Nakamura, and T. Inamura, "Humanoids walk with feedforward dynamic pattern and feedback sensory reflection," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 4, 2001, Conference, pp. 4220–4225.

[37] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi, and K. Tanie, "Planning walking patterns for a biped robot," *Robotics and Automation, IEEE Transactions on*, vol. 17, no. 3, pp. 280–289, June 2001.

[38] W. Huang, C. Chew, Y. Zheng, and G. Hong, "Pattern generation for bipedal walking on slopes and stairs," in *Humanoid Robots 2008, 8th IEEE-RAS International Conference on*, Dec 2008, Conference, pp. 205–210.

[39] S. Hwang, H. Kwak, and G. Park, "Stable walking of humanoid robot by using landing control in various environments," in *Control, Automation and Systems, 2007. ICCAS '07. International Conference on*, Oct 2007, Conference, pp. 1929–1932.

[40] O. Jenkins and M. Mataric, "Deriving action and behavior primitives from human motion data," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3, 2002, Conference, pp. 2551–2556.

[41] Y. Jun and P. Oh, "A 3-tier infrastructure: Virtual-, mini-, online-hubo stair climbing as a case study," *Proceeding Biomechanics and Robotics, ACTA Press*, vol. 752 Robotics, Nov 2011.

[42] Y. Jun, A. Alspach, and P. Oh, "Controlling and maximizing humanoid robot pushing force through posture," in *Ubiquitous Robotis and Ambient Intelligence (URAI), 2012 9th International Conferece on*, Nov 2012, Conference, pp. 158–162.

[43] Y. Jun and P. Oh, "'realiztion of miniature humanoid for obstacle avoidance with real-time zmp preview control used for full-sized humanoid," in *Humanoid Robots 2010, 10th IEEE-RAS International Conference on*, Dec 2010, Conference, pp. 46–51.

[44] Y. Jun, H. Wang, P. Oh, and Y. Zheng, "Stable walking toward real-world application," in *Robotics and Automation, 2014. Proceedings. 2014 IEEE International Conference on*, Nov 2014, Conference, pp. *peer–review.

[45] Y. Jun, J. Weisz, P. Oh, and P. Allen, "Posture control for humanoid manipulation on cluttered environments," in *Robotics and Automation, 2014. Proceedings. 2014 IEEE International Conference on*, Nov 2014, Conference, pp. *peer–review.

[46] Y. Jun, J. Weisz, C. Rasmussen, P. Oh, and P. Allen, "Real-time teleop with non-prehensile manipulation," in *Technologies for Practical Robot Applications (TePRA), 2014 IEEE International Conference on*, Apr 2014, Conference, pp. *peer–review.

[47] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation using preview control of the zero-moment-point," in *Robotics and Automation, 2003. Proceedings. 2003 IEEE International Conference on*, vol. 2, Sep 2003, Conference, pp. 1620–1626.

[48] ——, "Resolved momentum control: Humanoid motion planning based on the linear and angular momentum," in *Intelligent Robots and Systems, 2003. IEEE/RSJ International Conference on*, vol. 2, Oct 2003, Conference, pp. 1644–1650.

[49] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3d linear inverted pendulum model: A simple modeling for a biped walking pattern generation," in *Intelligent Robots and Systems, 2001. IEEE/RSJ International Conference on*, vol. 1, Oct 2001, Conference, pp. 239–246.

[50] S. Kajita, K. Yokoi, M. Saigo, and K. Tanie, "Balancing a humanoid robot using backdrive concerned torque control and direct angular momentum feedback," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 4, 2001, Conference, pp. 3376–3382.

[51] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, M. Hirata, K. Akachi, and T. Isozumi, "Humanoid robot hrp-2," in *Robotics and Automation, 2004. Proceedings. 2004 IEEE International Conference on*, vol. 2, Apr 2004, Conference, pp. 1083–1090.

[52] S. Kanzaki, K. Okada, and M. Inaba, "Bracing behavior in humanoid through preview control of impact disturbance," in *Humanoid Robots, 2005 5th IEEE-RAS International Conference on*, Dec 2005, Conference, pp. 301–305.

[53] T. Katayama, T. Ohki, T. Inoue, and T. Kato, "Design of an optimal controller for a discrete-time system subject to previewable demand," *International Journal of Control*, vol. 41, pp. 677–699, Mar 1985.

[54] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, vol. 2, May 1985, Conference, pp. 500–505.

[55] O. Khatib and J. Burdick, "Motion and force control of robot manipulators," in *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, vol. 3, Apr 1986, Conference, pp. 1381–1386.

[56] H. Kim, J. An, J. Moon, and D. Kim, "Simulation based for intelligent control system of multi humanoid robots for dynamic load carrying," in *Control Automation and Systems (ICCAS), 2010 International Conference on*, Oct 2010, Conference, pp. 101–104.

[57] J. Kim, "Walking pattern generation of a biped walking robot using convolution sum," in *Humanoid Robots 2007, 7th IEEE-RAS International Conference on*, Dec 2007, Conference, pp. 539–544.

[58] J. Kim and J. Oh, "Walking control of the humanoid platform khr-1 based on torque feedback control," in *Robotics and Automation, 2004. Proceedings. 2004 IEEE International Conference on*, vol. 1, April 2004, Conference, pp. 623–628.

[59] J. Kim, I. Park, and J. Oh, "Walking control algorithm of biped humanoid robot on uneven and inclined floor," *Journal of intelligent and robotic systems*, vol. 48, no. 4, pp. 457–484, Apr 2007.

[60] ——, "Realization of dynamic stair climbing for biped humanoid robot using force-torque sensors," *Journal of Intelligent and Robotic Systems*, vol. 56, pp. 389–423, 2009.

[61] M. Kim, S. Park, and J. Oh, "Realization of stretch-legged walking of the humanoid robot," in *Humanoid Robots 2008, 8th IEEE-RAS International Conference on*, Dec 2008, Conference, pp. 118–124.

[62] S. Kim and Y. Sankai, "Stair climbing task of humanoid robot by phase composition and phase sequence," in *Robots and Human Interactive Communication, IEEE International Workshop on*, 2005, Conference, pp. 531–536.

[63] Y. Kim, B. Lee, J. Yoo, J. Kim, and J. Ryu, "Compensation for the landing impact force of a humanoid robot by time domain passivity approach," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, May 2006, Conference, pp. 1225–1230.

[64] J. Kuffner and S. Lavalle, "Rrt-connect: An efficient approach to single-query path planning," in *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, vol. 2, 2000, Conference, pp. 995–1001.

[65] J. Kuffner, K. Nishwaki, S. Kagami, M. Inaba, and H. Inoue, "Motion planning for humanoid robots under obstacle and dynamic balance constraints," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 1, 2001, Conference, pp. 692–698.

[66] O. Kurt and K. Erbatur, "Biped robot reference generation with natural zmp trajectories," in *Advanced Motion control, 2006. 9th IEEE International Workshop on*, 2006, Conference, pp. 403–410.

[67] W. Kwon, H. Kim, J. Park, C. Roh, J. Lee, J. Park, W. Kim, and K. Roh, "Biped humanoid robot mahru-3," in *Humanoid Robots, 2007 7th IEEE-RAS International Conference on*, Nov 2007, Conference, pp. 583–588.

[68] S. Lee and A. Goswami, "Reaction mass pendulum (rmp): An explicit model for centroidal angular momentum of humanoid robots," in *Robotics and Automation, 2007 IEEE International Conference on*, Apr 2007, Conference, pp. 4667–4672.

[69] J. Li, Q. Huang, W. Zhang, Z. Yu, and K. Li, "Flexible foot design for a humanoid robot," in *Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on*, Sep 2008, Conference, pp. 1414–1419.

[70] T. Li, Y. Su, C. Kuo, C. Chen, C. Hsu, and M. Lu, "Stair-climbing control of humanoid robot using force and accelerometer sensors," in *SICE Annual Conference*, Sep 2007, Conference.

[71] H. Lim, S. Setiawan, and A. Takanishi, "Balance and impedance control for biped humanoid robot locomotion," in *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, vol. 1, 2001, Conference, pp. 494–499.

[72] K. Loffler, M. Gienger, F. Pfeiffer, and H. Ulbrich, "Sensors and control concept of a biped robot," *Industrial Electronics, IEEE Transactions on*, vol. 51, no. 5, pp. 972–980, Oct 2004.

[73] M. Mahani, S. Jafari, and R. Rahmatkhah, "Novel humanoid push recovery using knee joint," in *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, Oct 2011, Conference, pp. 446–451.

[74] W. Mao, J. Kim, and J. Lee, "Continuous steps toward humanoid push recovery," in *Automation and Logistics, 2009. ICAL '09. IEEE International Conference on*, Aug 2009, Conference, pp. 7–12.

[75] W. Mao, G. Qin, and J. Lee, "Humanoid push recovery strategy for unknown input forces," in *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, Aug 2009, Conference, pp. 1904–1909.

[76] H. Masato and T. Tooru, "Development of humanoid robot asimo," in *Honda Reaserch and Development Tech Rev*, vol. 13, 2001, Conference, pp. 1–6.

[77] S. McGill and D. Lee, "Cooperative humanoid stretcher manipulation and locomotion," in *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, Oct 2011, Conference, pp. 429–433.

[78] U. Nagarajan and A. Goswami, "Generalized direction changing fall control of humanoid robots among multiple objects," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, May 2010, Conference, pp. 3316–3322.

[79] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *Journal of Dynamic Systems, Measurement, and Control*, vol. 108, no. 3, p. 9, 1986.

[80] S. Nakaoka, A. Nakazawa, K. Yokoi, H. Hirukawa, and K. Ikeuchi, "Generating whole body motions for a biped humanoid robot from captured human dances," in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 3, Sep 2003, Conference, pp. 3905–3910.

[81] N. Nazir, S. Mitsuji, and N. Shigeki, "A siso linear system with an unstable inner loop and its application to zero moment point (zmp) control," in *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*, Oct 2006, Conference, pp. 2172–2177.

[82] N. Nazir, N. Shigeki, and S. Mitsuji, "Balance control analysis of humanoid robot based on zmp feedback control," in *Intelligent Robots and Systems, 2002, Proceedings. 2002 IEEE International Conference on*, vol. 3, 2002, Conference, pp. 2437–2442.

[83] K. Noh, J. Kim, and U. Huh, "Stability experiment of a biped walking robot with inverted pendulum," in *Industrial Electronics Society, 2004. IECON 2004. 30th Annual Conference of IEEE*, vol. 3, Nov 2004, Conference, pp. 2475–2479.

[84] S. Nozawa, Y. Kakiuchi, K. Okada, and M. Inaba, "Controlling the planar motion of a heavy object by pushing with a humanoid robot using dual-arm force control," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, May 2012, Conference, pp. 1428–1435.

[85] Y. Ogura, H. Aikawa, K. Shimomura, A. Morishima, H. Lim, and A. Takanishi, "Development of a new humanoid robot wabian-2," in *Robotics and Automation, 2006. Proceedings. 2006 IEEE International Conference on*, 2006, Conference, pp. 76–81.

[86] T. Ono, T. Murakami, and K. Ohnishi, "An apporach to biped robot control according to surface condition of ground," in *Advanced Motion Control, 1998. AMC'98-Coimbra, 1998 5th International Workshop on*, 1998, conference, pp. 129–134.

[87] C. Ott, D. Lee, and Y. Nakamura, "Motion capture based human motion recognition and imitation by direct marker control," in *Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on*, Dec 2008, Conference, pp. 399–405.

[88] H. Park, M. Ali, and C. Lee, "'convolution-sum-based generation of walking patterns for uneven terrains," in *Humanoid Robots 2010, 10th IEEE-RAS International Conference on*, Dec 2010, Conference, pp. 455–460.

[89] I. Park, J. Kim, and J. Oh, "Online biped walking pattern generation for humanoid robot khr-3(kaist humanoid robot-3: Hubo," in *Humanoid Robots 2006, 6th IEEE-RAS International Conference on*, Dec 2006, Conference, pp. 398–403.

[90] J. Park and Y. Youm, "General zmp preview control for bipedal walking," in *Robotics and Automation, 2007 IEEE International Conference on*, Apr 2007, Conference, pp. 2682–2687.

[91] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *Humanoid Robots, 2006 6th IEEE-RAS International Conference on*, Dec 2006, Conference, pp. 200–207.

[92] M. Raibert, K. Blankespoor, G. Nelson, and R. Playter, "Bigdog, the rough-terrain quadruped robot," in *In Proceedings of the 17th World Congress*, 2008, Conference.

[93] ROS. [Online]. Available: http://wiki.ros.org/rviz

[94] L. Rosenberg, "Virtual fixtures: Perceptual tools for telerobotic manipulation," in *Virtual Reality Annual International Symposium, 1993., 1993 IEEE*, 1993, Conference.

[95] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, "The intelligent asimo: System overview and integration," in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3, 2002, Conference, pp. 2478–2483.

[96] Schaft. [Online]. Available: http://www.dump.com/roboticschallenge

[97] L. Sentis and O. Khatib, "Synthesis of whole-body behaviors through hierarchical control of behavioral primitives," *International Journal of Humanoid Robotics*, vol. 2, no. 4, pp. 505–518, 2005.

[98] R. Sherbert and P. Oh, "Conductor: A controller development framework for high degree of freedom system," in *Robotics and Automation, 2011. Proceedings. 2011 IEEE International Conference on*, Sep 2011, Conference, pp. 1022–1029.

[99] A. Shon, K. Grochow, and R. Rao, "Robotic imitation from human motion capture using gaussian processes," in *Humanoid Robots, 2005 5th IEEE-RAS International Conference on*, Dec 2005, Conference, pp. 129–134.

[100] B. Siciliano and J. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Advanced Robotics, 1991. 'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on*, vol. 2, June 1991, Conference, pp. 1211–1216.

[101] B. Stephen, "Humanoid push recovery," in *Humanoid Robots, 2007 7th IEEE-RAS International Conference on*, Dec 2007, Conference, pp. 589–595.

[102] ——, "Integral control of humanoid balance," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, Oct 2007, Conference, pp. 4020–4027.

[103] M. Stilman, K. Nishiwaki, and S. Kagami, "Learning object models for whole body manipulation," in *Humanoid Robots, 2007 7th IEEE-RAS International Conference on*, Nov 2007, Conference, pp. 174–179.

[104] ——, "Humanoid teleoperation for whole body manipulation," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, May 2008, Conference, pp. 3175–3180.

[105] J. Stuckler and S. Behnke, "Following human guidance to cooperatively carry a large object," in *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, Oct 2011, Conference, pp. 218–223.

[106] Y. Sugahara, T. Hosobata, Y. Mikuriya, H. Lim, and A. Takanishi, "Realization of stable dynamic walking by a parallel bipedal locomotor on uneven terrain using a virtual compliance control," in *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 1, Oct 2003, Conference, pp. 595–600.

[107] Y. Takahashi, Y. Uchiyama, S. Shigeta, and H. Kameyama, "Impact relaxation control system of jumping humanoid robot," in *Control, Automation and Systems, 2007. ICCAS '07. International Conference on*, Oct 2007, Conference, pp. 2678–2682.

[108] T. Tsujita, A. Konno, S. Komizunai, Y. Nomura, T. Owa, T. Myojin, Y. Ayaz, and M. Uchiyama, "Analysis of nailing task motion for a humanoid robot," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, Sep 2008, Conference, pp. 1570–1575.

[109] C. Urmson and W. Whittaker, "Self-driving cars and the urban challenge," *Intelligent Systems, IEEE*, vol. 23, no. 2, pp. 66–68, Mar 2008.

[110] M. Vukobratovic and B. Borovac, "Zero-moment point-thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, pp. 157–173, Dec 2004.

[111] M. Vukobratovic and J. Stepanenko, "On the stability of anthropomorphic systems," *Mathematical Biosciences*, vol. 15, no. 1, pp. 1–37, Oct 1972.

[112] H. Wang, Y. Jun, Y. Zheng, and P. Oh, "Drc-hubo walking on rough terrains," in *Technologies for Practical Robot Applications (TePRA), 2014 IEEE International Conference on*, Apr 2014, Conference, pp. *peer–review.

[113] J. Yamaguchi, N. Kinoshita, A. Takanishi, and I. Kato, "Development of a dynamic biped walking system for humanoid development of a biped walking robot adapting to the humans' living floor," in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 1, Apr 1996, Conference, pp. 232–239.

[114] K. Yamane and J. Hodgins, "Simultaneous tracking and balancing of humanoid robots for imitating human motion capture data," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, Oct 2009, Conference, pp. 2510–2517.

[115] S. Yi, B. Zhang, D. Hong, and D. Lee, "Online learning of a full body push recovery controller for omnidirectional walking," in *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, Oct 2011, Conference, pp. 1–6.

[116] K. Yokoyama, H. Handa, T. Isozumi, Y. Fukase, K. Kaneko, F. Kanehiro, F. Tomita, and H. Hirukawa, "Cooperative works by a human and a humanoid robot," in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 3, Sep 2003, Conference, pp. 2985–2991.

[117] P. Zhang, C. Zhou, L. Zhang, Y. Tian, and Z. Liu, "Adaptive compliant control of humanoid biped foot with elastic energy storage," in *Advanced Intelligent Mechatronics, 2009. AIM 2009. IEEE/ASME International Conference on*, July 2009, Conference, pp. 928–933.

[118] Y. Zheng and J. Shen, "Gait synthesis for the sd-2 biped robot to climb sloping surface," *Robotics and Automation, IEEE Transaction on*, vol. 6, no. 1, pp. 86–96, 1990.

[119] Y. Zheng, S. Wang, Y. Li, D. Orin, K. Sohn, Y. Jun, and P. Oh, "Planning complex physical tasks for disaster response with a humanoid robot," in *Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference on*, Apr 2013, Conference, pp. 1–6.

[120] M. Zucker, Y. Jun, B. Killen, T. Kim, and P. Oh, "Continuous trajectory optimization for autonomous humanoid door opening," in *Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference on*, Apr 2013, Conference, pp. 1–5.

Appendix A. 3-TIER INFRASTRUCTURE: RESEARCH TOOL AND PLATFORM

In past two decades a lot of humanoid research papers have been published. Many researchers have simulated or built their own humanoid robots to verify and validate algorithms and control techniques. However, the current actual humanoid technology has not caught up with the achievements in such papers since there was a critical gap; platforms were not easily available, and mechanically and systematically different, and hence research was difficult to validate and hence results were hard to reproduce.

In 2007 the National Science Foundation in the United States awarded a grant that partnered American and Korean roboticists to create a 3-tier infrastructure to stimulate and advance humanoid research[1]. This infrastructure served to fill such critical gaps. To overcome this challenge, virtual-, mini- and online-Hubo were constructed to respectively prototype, test-and-evaluate, and verify-and-validate humanoid-based research. Based on the full-sized KAIST Hubo humanoid, virtual-Hubo is a zero-to-low cost simulator, mini-Hubo is a 17-inch scaled open-source robot that one can construct for less than 10,000 USD and online-Hubo is the full-sized robot (called Jaemi Hubo) that is tethered to a gantry and can be programmed and monitored using the Internet in order to be world-widely accessed. An integrated software called 'Conductor' [98], was developed to access 3-tier infrastructure and enabled a wide range of investigators to gain entry into humanoid research without being overly hindered by cost or maintenance.

A.1   Virtual-Hubo

Virtual-Hubo is an emulator component for the 3-tier infrastructure. (Figure A.1) Open Dynamics Engine (ODE)-based software allows one to better visualize how motion algorithms perform. A graphical environment allows one to rapidly prototype and visualize algorithms that will be applied to mini- and online-Hubo. This emulator can import CAD models and employs the Webot ODE. The platform-independent architecture allows algorithms that run on virtual-Hubo to port seamlessly to mini- and online-Hubo. The platform-independent architecture, called Conductor [98], was developed by our group in order to allow algorithms that run on virtual-Hubo to port seamlessly to mini- and online-Hubo. This is accomplished by representing the control elements of the system in terms of a state variable approach, which is easily adjustable between the three architectures. The elements of control which are platform specific (motor drivers, calibration, etc) are left to lower level drivers that are easily interchangeable.

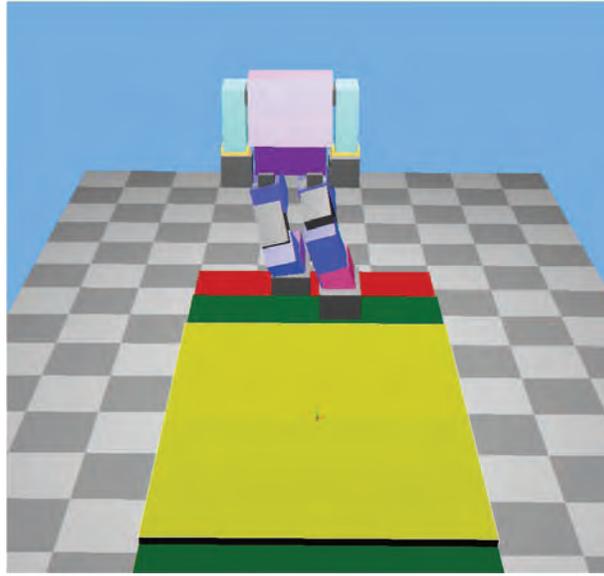[1]This project was supported by a US NSF Grant 0730206

Figure A.1: Virtual simulation environment for rapid prototyping, Virtual-Hubo

A.2 Mini-Hubo

Under the NSF grant that supports work in this project, the authors' collaborator (Virgina Tech's RoMeLa Lab) design mini-Hubo (Figure A.2), a 22 DOF scaled version of the KAIST Hubo, which stands 0.5 m tall and weighs 3 kg. The mini-Hubo was designed to be open-sourced; plans for anyone to construct this robot are freely available on the web. Estimated construction costs are under 10,000 USD with less than 40 person-hours of machining and assembly time [43].

The key design considerations of the miniature humanoid are to minimize weight, have strong actuators to produce demanded forces, and centralize mass in the body. Minimization of weight is most critical, because it reduces power and torque requirements to perform a given motion. The cost of adding weight is stronger (and heavier) actuators, which in turn requires larger batteries and support structure.

Our mechanical design for mini-Hubo considered these issues in implementation. The mini-Hubo Dynamixel RX28 digital servos are widely used in many humanoid research labs. Weighing only 72g, the servo generates 28.3 kg-cm holding torque, with adjustable control gains. The total weight of mini-Hubo is 3.0 Kg including only 550g weight of brackets. By centering the Lithium Polymer (LiPo) battery directly above the hip joint, the center of mass is balanced in the sagittal plane.

The Mini-Hubo has 22 Degrees Of Freedom (DOF) total in its arms and legs. The six total DOF in each leg allow very similar motion to human legs. Eight ultrasonic sensors around the torso and a 320*240 pan/tilt CMOS camera in the head provide data used for obstacle avoidance.

The system architecture of mini-Hubo follows that of Hubo [58]. We utilize Fit

Figure A.2: Custom designed miniature humanoid robot for testing-and-evaluating, Mini-Hubo

PC for the main computer which is a small and portable computer involving Inter Atom 1.6Ghz processor. The main computer is accessed by remote control based on Window XP. Each dynamixel servo and CMOS camera communicate with the main computer via RS-485 and USB respectively. The Visual Studio 2009 and MATLAB were used as a main program to process the data and to generate all trajectories for the motion of mini-Hubo.

A.3 Online-Hubo

Hubo is a full-sized humanoid robot and open-platform developed in KAIST. (Figure A.3) It is 130cm tall and 42kg weight without the battery. In terms of mechanics, Hubo has 38 Degree Of Freedom (DOF) including 12 DOF in legs, 12 DOF in arms, 10 DOF in hands, 1 DOF on torso, and 3 DOF on neck. Each joint is driven by a BLDC motor with harmonic drive and has an optical encoder that allows user to control each joint angle. There are four 3-axis Force-Torque (F/T) sensors located on each wrist and ankle joint, and 1 Inertial Measurement Unit (IMU) on the center of pelvis. In system architecture, two main computers are located on the chest. One dedicates joint control and sensor measurement via CAN protocol and the other contributes vision sensing and image process. Real-Time Operating system and customized control boards are the main feature of Hubo which compensate the delay and interpolate the error so that user can control each actuator with hardware-based frequency [58]. The Hubo robot has two robot hands. In Hubo hands, five fingers are placed on each hand and a small DC motor operates each finger where three finger
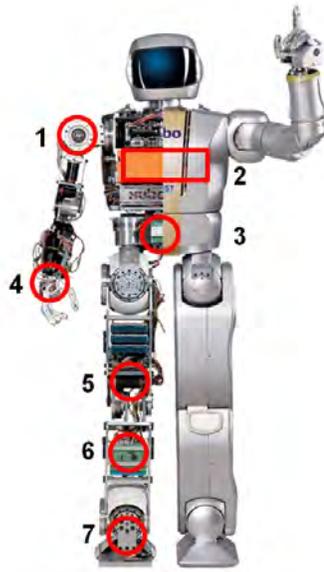
Figure A.3: A full-sized humanoid robot for verifying-and-validating, Online-Hubo: 1)harmonic driver, 2) two computers for body control and high computational processing, and battery, 3) Inertial Measurement Unit, 4) Force-Torque sensor on wrist, 5) Bruchless DC motor, 6) motor control board, and 7) Force-Torque sensor on ankle

joints are connected through the cable. Current net force in grasp is 1.5 kg.

Online-Hubo is built on Hubo. The combination of remotely accessible and controllable software, and a powered safety harness allows the Hubo to be operated remotely [98]. Researchers can access the robot anywhere, and monitor its motion via the Internet in real-time. With the Conductor, a program demonstrated in Virtual-Hubo can be immediately loaded to online-Hubo to verify its performance.

A.4   DRC-Hubo

The DRC-Hubo (Figure A.4 is the latest version of the Hubo robot series developed by Rainbow Inc and KAIST in South Korea. It is designed to perform the practical tasks like driving a car, walking on rough terrain, climbing a ladder, breaking a wall, removing debris, turning a valve, opening a door, and grasping/dragging a hose in human-centered environments. Compared to previous versions, the arms and legs have been significantly lengthened to increase the robots workspace. The maximum joint torque for each joint has been increased to approximately twice that of the previous version, hubo2 [58], to allow for handling tools and heavier objects. The DRC-Hubos seven Degree-Of-Freedom (DOF) arms provide kinematic redundancy which brings more reliable motion in manipulation.

DRC-Hubo is 1.2m tall without a head and weighs 48kg without a battery. Each joint has a built in PD controller for position control. The pelvis of the robot contains
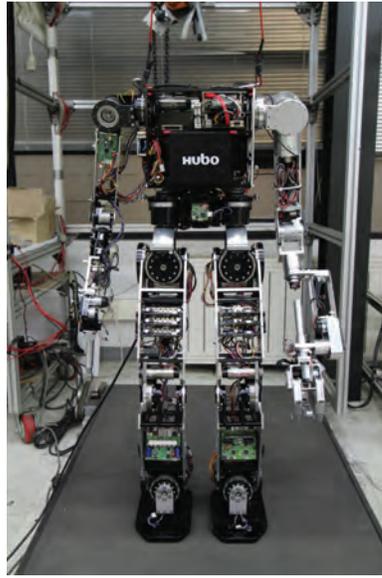
Figure A.4: DRC-Hubo

a three-axis Inertial Measurement Unit (IMU). The IMU measures angular variation in roll, pitch and yaw of the body. A three-axis F/T sensor is located in each of the ankles and wrists. These F/T sensors are used to measure the reaction forces and moments from environmental contacts. This sensor data is used for balance and compliance control in walking and manipulation. There is also a floor facing camera on the bottom of the torso for reviewing foot positions.