

Hierarchical Optimization for Whole-Body Control of Wheeled Inverted Pendulum Humanoids

Munzir Zafar, Seth Hutchinson, Evangelos A. Theodorou

Abstract—In this paper, we present a whole-body control framework for *Wheeled Inverted Pendulum (WIP) Humanoids*. WIP Humanoids are redundant manipulators dynamically balancing themselves on wheels. Characterized by several degrees of freedom, they have the ability to perform several tasks simultaneously, such as balancing, maintaining a body pose, controlling the gaze, lifting a load or maintaining end-effector configuration in operation space. The problem of whole-body control is to enable simultaneous performance of these tasks with optimal participation of all degrees of freedom at specified priorities for each objective. The control also has to obey constraint of angle and torque limits on each joint. The proposed approach is hierarchical with a low level controller for body joints manipulation and a high-level controller that defines center of mass (CoM) targets for the low-level controller to control zero dynamics of the system driving the wheels. The low-level controller plans for shorter horizons while considering more complete dynamics of the system, while the high-level controller plans for longer horizon based on an approximate model of the robot for computational efficiency.

I. INTRODUCTION

Wheeled inverted pendulum systems offer the best of two worlds. First, their wheels make for inherently fast and efficient locomotion—something that bipedal system designers are still struggling to achieve. Second, the dynamically balancing inverted pendulum endows them with the ability to deal with very heavy payloads, their torques being canceled by a readily adjustable CoM—something that statically stable wheeled platforms can not achieve. These characteristics make them attractive for a wide range of applications such as Segway human transporters [1]–[5], transporters with seats [4]–[8], self-balancing wheel chairs [9]–[12] and WIP Humanoids [13]–[21]. WIP Humanoids add to the abilities of a WIP by offering a redundant manipulator, with one or more arms, that can be controlled to intelligently interact with their environment and perform useful tasks. If they could be controlled for safely handling large forces, their ability to dynamically cancel out their effects would prove useful for assisting humans with tasks requiring large effort. The WIP humanoid in [18] was designed with this purpose in mind. Its two-arm-bearing torso is mounted on massive base and spine links, thus providing it with the ability to perform heavy tasks by manipulating its weight torque as needed.

The challenge in controlling such robots is their nonlinear, highly unstable and under-actuated dynamics. A large body

Munzir Zafar, Seth Hutchinson and Evangelos A. Theodorou are with the Institute of Robotics and Intelligent Machines at the Georgia Institute of Technology, Atlanta, GA, 30332, USA. email: mzafar7@gatech.edu, seth@gatech.edu, evangelos.theodorou@gatech.edu

of literature exists to deal with these typical problems of WIP systems [3], [6], [20], [22]–[36]. However, the focus of most studies remains a simplified system having one-link attached to the wheels. This has inspired some to leverage this work for WIP humanoid control, by treating the control of wheeled inverted pendulum independently from the control of upper body—the former being a simplified model of the full robot with one link of an equivalent center of mass (CoM). This technique was also utilized to control the WIP humanoid system in [37].

The problems associated with such a treatment become apparent when precision-critical or safety-critical tasks are performed. For example, since forward motion is left up to the wheels alone, and the upper body is not compensating for pitch changes induced by the demands of locomotion on the wheel controller, the end-effector can hardly obey constraints on its position and orientation during locomotion, which may sometimes be critical to perform a task. Similarly, when dealing with sudden changes in large forces, the transients induced on a balancing controller due to the switching of equilibrium positions can not be properly managed if the body is blind to the control of its center of mass. This makes the case for a unified approach to locomotion and manipulation tasks.

In this work we propose a unified framework that deals with whole-body control of WIP humanoid structures such as Golem Krang [18], and we aim to demonstrate the application of the proposed solution on this platform. In particular our core contributions are:

- Deriving the full 3D dynamic model of a typical WIP Humanoid
- Developing a hierarchical control framework in which different tasks can be performed on the WIP platform in a unified fashion resulting in a better overall performance
- Showing how quadratic programming (QP) based control techniques [38] can be applied as a low-level controller to isolated manipulator dynamics obtained by elimination of wheel dynamics from the dynamic model
- Analysing the zero dynamics of the system to motivate the use of a simplified Wheeled Inverted Pendulum Model (WIPM) that approximates the inertial properties of the full robot for predictive control by a high-level controller
- Using differential dynamic programming (DDP) and model predictive control (MPC) for generating and controlling CoM trajectory of the WIP Humanoid using predictions of the WIPM. This serves as the high-level

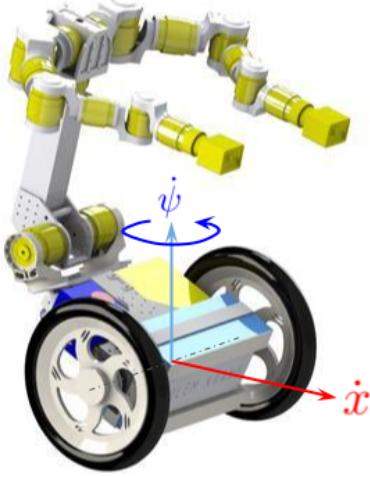


Fig. 1: A Wheeled Inverted Pendulum Humanoid. 3D Simulation video provided with this paper applies the proposed approach on this robot.

controller for providing feasible CoM targets to the low-level controller to ensure effective balancing and locomotion along with the performance of other tasks

A brief outline of this document is as follows: Section II describes the dynamic model of the system under consideration. Section III gives complete description of the hierarchical approach for whole body control of the robot. Results on a 7-DOF planar robot are presented in section IV. They represent an illustrative scaled down version of a 19-DOF 3D simulation video submitted with this paper. This is followed by conclusion in section V.

II. MODELING

In this section, we present the dynamic model of a WIP humanoid under consideration [18]. The techniques usually utilized for dynamic modeling of a WIP system are Euler-Lagrange, Newton's laws of motion or Kane's method. Assumptions include rigid bodies, flat ground and zero slip. Another simplification for dynamic modeling is to restrict motion to the sagittal plane [23], [39]–[45]. Our analysis does not impose this restriction, and our robot is allowed to explore the full range of accessible workspace. Others have explored this approach [22], [46]–[49]; however, they perform their analysis on simple WIP systems, with the body consisting of either just one link, or a simplifying assumption is used to treat it as such. Our analysis does not make any such simplifying assumptions in the derivation of the model.

The system under consideration is shown in Figure 1. It is a highly redundant manipulator mounted on a differential wheel drive able to dynamically balance itself in an inverted pendulum configuration. We are interested to control the robot that is assumed to be always moving on a horizontal ground, thus limiting its DOFs in free space (*i.e.*, as opposed to the base floating in the air). However, we begin with defining a set of coordinates that specify all its DOFs that are able to model the configuration when the base is floating in

the air. We do this in order to utilize off-the-shelf simulation tools (*e.g.* DART [50]) that make use of fast algorithms like Featherstone's [51] for computation of quantities required by our control algorithm, such as the inertia matrix. We use the term “body” to refer to the entire structure of the robot excluding the wheels and the term “base link” to identify the first link of the body attached to the wheels. Assuming n is the total number of links in the body, the full configuration of the robot can be represented using $7 + n$ coordinates. These include 6 DOFs of the floating base, 2 angular positions of the wheels relative to the base link, and $n - 1$ motions of the links in the body relative to their respective parent links. Applying Euler-Lagrange to derive the dynamic model results in

$$A_{full}\ddot{q}_{full} + C_{full}\dot{q}_{full} + Q_{full} = [0_{1 \times 6} \quad \tau^\top]^\top \quad (1)$$

where $q_{full} \in \mathbb{R}^{7+n}$ is the set of generalized coordinates; $\tau \in \mathbb{R}^{1+n}$ is the vector of torques comprising 2 wheel torques (τ_L and τ_R) along with $n - 1$ body joint torques; A_{full} , C_{full} and Q_{full} are respectively the inertia matrix, Coriolis matrix and gravity vector.

Since we consider the case when this robot is moving on a horizontal ground under non-holonomic constraints, the degrees of freedom of the system reduce to $3 + n$, which include: (1) robot heading \dot{x} (Fig 1), (2) robot spin $\dot{\psi}$ (Fig 1), (3) pitch of the base link with respect to the vertical \dot{q}_1 (Fig. 2b) and (4) motions of the $n - 1$ joints mounted on the base link $[\dot{q}_2 \dots \dot{q}_n]^\top$ (Fig 2b). So the minimum set of coordinates are $\dot{\mathbf{q}} = [\dot{x} \quad \dot{\psi} \quad \dot{q}^\top]^\top$ where we define $q = [q_1 \dots q_n]^\top$. Define J_{tf} as the transform Jacobian relating full set of coordinates and minimum set of coordinates

$$\dot{q}_{full} = J_{tf}\dot{\mathbf{q}} \quad (2)$$

The exact expression for J_{tf} is dependent on the choice of reference frames and is trivially found as a function of q_1 . Space limitation precludes a detailed account. The dynamic model in minimum set of coordinates can be obtained as¹

$$A\ddot{\mathbf{q}} + C\dot{\mathbf{q}} + Q = B\tau \quad (3)$$

where $A = J_{tf}^\top A_{full} J_{tf}$, $C = J_{tf}^\top A_{full} \dot{J}_{tf} \dot{\mathbf{q}} + J_{tf}^\top C_{full}$, $Q = J_{tf}^\top Q_{full}$, $B\tau = J_{tf}^\top ([0_{1 \times 6} \quad \tau^\top]^\top)$. Here $B \in \mathbb{R}^{(n+2) \times (n+1)}$ is the actuation matrix. Its dimensions indicate that there is one less actuator in the system than the total degrees of freedom. B is defined as

$$B = \begin{bmatrix} \bar{B} & 0_{3 \times (n-1)} \\ 0_{(n-1) \times 2} & I_{n-1} \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} \frac{1}{\bar{R}} & \frac{1}{\bar{L}} \\ -\frac{L}{2\bar{R}} & \frac{L}{2\bar{R}} \\ -1 & -1 \end{bmatrix} \quad (4)$$

As for actuation torques, the following points are noteworthy:

- 1) Forward motion \dot{x} is actuated by sum of wheel torques

¹Note that \dot{x} is a quasi-velocity [52] and therefore requires formulations other than Euler-Lagrange for correct derivation. We have originally derived (3) using Kane's formulation and noticed that the terms that are not captured in the presented treatment have negligible dynamic effect (they are function of $\dot{\psi}$ ²). We therefore omitted Kane's analysis for brevity.

- 2) Spin motion $\dot{\psi}$ is actuated by the difference of wheel torques
- 3) The base joint rotation q_1 is actuated by the reaction torque on wheel motors fixed on the base and driving the wheels
- 4) The rest of the joints in the tree structure are actuated by their respective joint torques

a) *Under-actuated System:* Wheel torques τ_L and τ_R control three motions: \dot{x} , $\dot{\psi}$ and \dot{q}_1 . Define

$$\begin{aligned}\tau_1 &= -(\tau_L + \tau_R) \\ \tau_0 &= \frac{L}{2R}(\tau_L - \tau_R)\end{aligned}\quad (5)$$

τ_0 actuates $\dot{\psi}$, while τ_1 actuates both \dot{x} and \dot{q}_1 . This is one of two types of under-actuation that can feature in WIP systems. In this type, reaction torque of the wheel motors acts on the base link as the motors are mounted on the link. In the second type, the base link is connected to the wheel cart via a passive joint, so the base link experiences no direct actuation. The first type is easier to build but more difficult to control. Most literature on WIP systems however focuses on the second type of systems (also noted by [53]). We focus on the first type of systems.

III. HIERARCHICAL CONTROL

The specific problem of whole body control of WIP humanoids is addressed by Toshiyuki *et al.* [14]–[16]. Their work differs from ours in four important aspects: In the system we consider, the base-link is experiencing reaction torque of the wheels while they consider only the case of a passive base joint. They restrict the angular motion of this joint whereas we allow this joint to participate in body manipulation. Their control treatment is restricted to the sagittal plane, however, we present a framework applicable to 3D manipulations. Finally, their control approach is based on inverse kinematics, as opposed to our formulation where we use inverse dynamics that lends itself easily to compliant motions and force interactions.

Figure 2a gives an overview of the proposed approach. It is hierarchical with a low level controller responsible for controlling the manipulator/body and a high-level controller that defines center of mass (CoM) targets for the low-level controller to control zero dynamics of the system driving the wheels. The low-level controller plans for shorter horizons while considering more complete dynamics of the system, while the high-level controller plans for longer horizon based on an approximate model of the robot for computational efficiency.

A. Low-Level Controller

With the definitions of τ_0 and τ_1 in (5) and defining $\Gamma = [\tau_1 \dots \tau_n]$, (3) can be written as

$$A \begin{pmatrix} \ddot{x} \\ \ddot{\psi} \\ \ddot{q} \end{pmatrix} + C \begin{pmatrix} \dot{x} \\ \dot{\psi} \\ \dot{q} \end{pmatrix} + Q = \begin{pmatrix} -\tau_1/R \\ \tau_0 \\ \Gamma \end{pmatrix} + \Gamma_{fric}, \quad (6)$$

Note that $\Gamma_{fric} \in \mathbb{R}^{(n+1) \times 1}$ represents the frictional effects (which we had omitted in previous equations for notational simplicity). In (6), we can easily observe that both \dot{x} and \dot{q}_1 are actuated by the same torque τ_1 .

1) *Isolating Manipulator Dynamics:* In (6), wheel and manipulator dynamics are coupled, not only by the wheel torque τ_1 , but also the inertia matrix A . In [54], elimination of \ddot{x} is performed in this system to determine a direct relationship between manipulator joint accelerations \ddot{q} and torques Γ . The inertia matrix is first defined in block form

$$A = \begin{bmatrix} a_{xx} & a_{x\psi} & a_{xq}^\top \\ a_{x\psi} & a_{\psi\psi} & a_{\psi q}^\top \\ a_{xq} & a_{\psi q} & A_{qq} \end{bmatrix},$$

Then inverting this matrix, followed by some algebraic manipulations, results in the following ODEs that isolate the manipulator dynamics from wheel dynamics

$$A\ddot{q} + P \left(C \begin{pmatrix} \dot{x} \\ \dot{q} \end{pmatrix} + Q - \Gamma_{fric} \right) = \Gamma, \quad (7)$$

where:

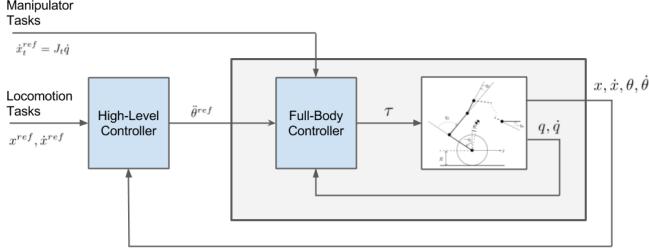
$$\begin{aligned}\mathcal{A} &= (I - \beta B) A_{qq}^*, \quad P = (I - \beta B) \left(-\frac{a_{xq}}{a_{xx}} I \right), \\ A_{qq}^* &= A_{qq} - \frac{1}{a_{xx}} a_{xq} a_{xq}^\top, \quad B = \left[\frac{a_{xq}}{Ra_{xx}} \quad 0_{n \times (n-1)} \right].\end{aligned}$$

The terms β, α are defined as $\beta = \frac{1}{1+\alpha}, \alpha = \frac{a_{xq}}{Ra_{xx}}$ where $a_{xq} = \text{first element of } a_{xq}$. The matrix \mathcal{A} is asymmetric. Expression for the new matrices \mathcal{A} and P that appear in (7) are derived in terms of block elements of the original inertia matrix A so there is no need to invert this matrix during run time to find the required matrices. Using the set of ODEs (7) it is now possible to apply full body control techniques to attain manipulator objectives.

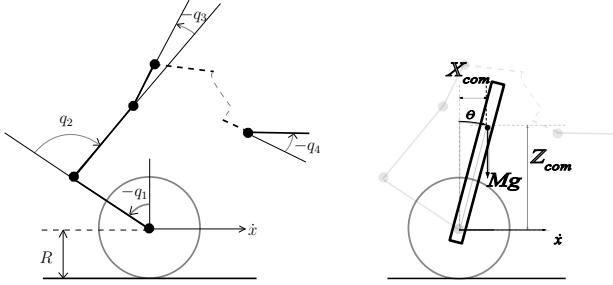
2) *Optimization Based Control:* The use of inverse dynamics (ID) and inverse kinematics (IK), in conjunction with optimization programs minimizing a single-step cost set up to perform a desired task in the operational space, has gained popularity recently for whole-body control. We prefer the work of [38] because it offers flexibility in terms of incorporating constraints such as joint, position and torque limits, obstacle avoidance and allows prioritization among tasks. The minimization variables X can be joint torques, accelerations, speeds or contact wrenches. Dynamics are linear in joint accelerations. Similarly, kinematics are linear in joint speeds. Thus constraints involving joint motion will be linear, allowing fast Quadratic Programming (QP) algorithms to solve the optimization. The optimization program is set up as a quadratic cost as follows:

$$\begin{aligned}\min & \left[\frac{1}{2} X^\top G X + g^\top X \right] \\ \text{s.t. } & C_E X + c_E = 0, \quad \text{and} \quad C_I X + c_I \leq 0.\end{aligned}$$

The cost function takes the form $0.5 \|P X - b\|^2$ so $G = P^\top P$ and $g = -P^\top b$. An example of cost function designed to perform task x_t (with Jacobian J_t), will have $X = \dot{q}$, $P = J_t$, $b = \ddot{x}_t^* - J_t \dot{q}$ and $\ddot{x}_t^* = K_p(x_t^{ref} - x_t) + K_d(\dot{x}_t^{ref} -$



(a) Overview of the proposed approach



(b) A Wheeled Inverted Pendulum (c) Wheeled Inverted Pendulum Model Robot with n joints

Fig. 2

$\dot{x}_t + \ddot{x}_t^{ref}$. The dynamics of manipulator (7) will be used as the equality constraint to incorporate torque limits. Then performing inverse dynamics on (7) with the optimal \ddot{q} obtained from QP will generate the resulting joint torques Γ . For prioritizing among multiple tasks, P and b can be re-written as [38]:

$$P^T = [w_o P_o, \dots w_n P_n], b^T = [w_o b_o \dots w_n b_n]. \quad (8)$$

A set of desirable robot behaviors are specified via this cost function as per the goal of the high level controller with higher penalties w_i assigned for higher priority tasks. A summary of the forms P and b matrices will take for various objectives appears in [38].

B. High-Level Controller

As previously noted, wheel motor torque is also driving the base link. The low-level controller, proposed in the previous subsection, utilizes this torque to directly control only the upper body motions ignoring the wheel dynamics that result from it. This is related to the zero dynamics of the system. The high-level controller proposed in this subsection therefore controls the wheels by planning for body CoM. This is set up as a trajectory optimization problem where a quadratic cost in terms of tracking error is to be minimized. To close the loop for dealing with deviations from optimal path due to unmodeled dynamics, we perform model predictive control (MPC) at sampling intervals T_s for a short horizon $t_H = NT_s$. MPC performs trajectory optimization on a continuous basis over t_H on an updated version of the simplified model, and its first control value is

used as the reference for low-level controller QPs over the next sampling period T_s .

We use differential dynamic programming (DDP) for trajectory optimization over the full horizon, as well as, optimization in every MPC iteration over the smaller horizon t_H . See [55] for the historical presentation and [56] for a modern treatment of DDP.

1) *Zero Dynamics*: We now derive the zero dynamics of the full robot and present an analysis motivating the use of CoM trajectory generation for controlling the zero dynamics of the system. We find the zero dynamics of the robot by eliminating τ_1 from the \dot{x} and \dot{q}_1 equations in the dynamic model. The two equations of the dynamic model are

$$\begin{pmatrix} a_{xx} & a_{xq}^\top \\ a_{xq} & a_{qq_1}^\top \end{pmatrix} \begin{pmatrix} \ddot{x} \\ \ddot{q} \end{pmatrix} + \begin{pmatrix} C'_x \\ C'_{q_1} \end{pmatrix} + \begin{pmatrix} Q_x \\ Q_{q_1} \end{pmatrix} = \begin{pmatrix} -\frac{\tau_1}{R} \\ 0 \end{pmatrix}, \quad (9)$$

where a_{qq_1} is the first row of A_{qq} , $[C'_x \ C'_{q_1}]^\top$ and $[Q_x \ Q_{q_1}]^\top$ are the first two elements of C' and Q respectively. Here we have lumped together Coriolis and frictional effects into C' matrix as:

$$C' = C \begin{pmatrix} \dot{x} \\ \dot{q} \end{pmatrix} - \Gamma_{fric}.$$

Eliminating τ_1 from the two equations results in:

$$\begin{aligned} & (Ra_{xx} + a_{xq_1})\ddot{x} + (Ra_{xq}^\top + a_{qq_1}^\top)\ddot{q} + (RC'_x + C'_{q_1}) + (RQ_x + Q_{q_1}) = 0 \\ & \Rightarrow \left[R \left(2m_w + \frac{2I_w}{R^2} + M \right) + MZ_{com} \right] \ddot{x} \\ & + \left[R \begin{pmatrix} MZ_{com} \\ M_2 Z_{com(2)} \\ \vdots \\ M_n Z_{com(n)} \end{pmatrix}^\top + \begin{pmatrix} I \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix}^\top \right] \ddot{q} + (RC'_x + C'_{q_1}) - MgX_{com} = 0, \end{aligned} \quad (10)$$

where:

m_w, I_w : are wheel mass and inertia about wheel-axis

M : is full body mass.

I : is full body inertia around wheel axis.

$[X_{com}, Z_{com}]$: are coordinates of body CoM.

M_j : is the mass of the articulated structure on joint i defined as $M = \sum_{k=j}^n m_k$.

$Z_{com(j)}$: is the z coordinate of CoM of the articulated structure about the wheel axis.

β_j : capture coupling torques on wheel axis due to the motion of the articulated structure on joint i

In deriving (10), we have utilized expressions for a_{xx} , a_{xq_1}, a_{xq} , a_{qq_1} , Q_x and Q_{q_1} obtained by symbolic evaluation of (6) while neglecting effects of spin motion ($\dot{\psi}$ and $\ddot{\psi}$) (space limitations preclude a more detailed account). Note that in the absence of transient dynamic forces (*i.e.*, when $\ddot{q} = \dot{q} = 0$) the only term driving \dot{x} motion is the torque due to body weight MgX_{com} . Modeling the CoM motion ($\dot{\theta}, \ddot{\theta}$), does capture some effects of the joint motions (\dot{q}, \ddot{q}). This provides a strong motivation to approximate the full robot as a simplified model—a single link of an equivalent CoM and inertia attached to the wheels—for the purpose of fast trajectory planning to dictate the zero dynamics of the robot for controlling wheel dynamics.

2) *Simplified Model*: The high-level planner we propose sees the robot as a simplified model that approximates the overall dynamics. This approach is similar to [38], where they approximate the bipedal humanoid with a Linear Inverted Pendulum Model (LIPM). We propose to use a Wheeled Inverted Pendulum Model (WIPM). The WIPM is a one-link robot dynamically balancing itself on the wheel (Figure 2c). Symbolic evaluation of (3) for the case of single link in the body gives us WIPM dynamics. We use θ to represent the pitch of the link. For the full robot, θ is the angular position of the body CoM. Using \dot{x} and \dot{q}_1 equations in (3), the model of WIPM is:

$$R\left(m_w + \frac{I_w}{R^2} + M\right)\ddot{x} + RMZ_{com}\ddot{\theta} + RMX_{com}\dot{\theta}^2 = -\tau_1 \quad (11)$$

$$MZ_{com}\ddot{x} + I\ddot{\theta} - MX_{com}g = \tau_1. \quad (12)$$

The low-level controller described earlier is used to control θ . High-level controller is designed to control the horizontal motion \dot{x} using θ . So it plans for a trajectory of θ in order to control \dot{x} . Eliminating τ_1 from (11) and (12) gives us the zero dynamics of the simplified model:

$$\begin{aligned} & \left[R\left(m + \frac{I_w}{R^2} + M\right) + MZ_{com}\right]\ddot{x} + (MZ_{com} + I)\ddot{\theta} \\ & + RMX_{com}\dot{\theta}^2 - MgX_{com} = 0. \end{aligned} \quad (13)$$

To obtain the simplified model from the full model, parameters R , m_w , I_w , x and \dot{x} will be directly used. Mass of the full robot is found using $M = \sum_{i=1}^n m_i$ and remains constant. Other parameters $\lambda(q) = (X_{com} \ Z_{com} \ \theta \ \dot{\theta} \ I)$ are function of full state q and will be obtained as follows:

$$\begin{bmatrix} X_{com} \\ Z_{com} \end{bmatrix} = \frac{1}{M} \sum_{i=1}^n m_i \begin{bmatrix} x_{com(i)} \\ z_{com(i)} \end{bmatrix}, \quad \theta = \arctan\left(\frac{X_{com}}{Z_{com}}\right)$$

$$\dot{\theta} = \frac{\cos\theta}{Z_{com}} [\cos\theta \ -\sin\theta] J_{com}\dot{q}, \quad I = a_{(q1)(q1)},$$

where $x_{com(i)}$ and $z_{com(i)}$ are CoM coordinates of link i , $a_{(q1)(q1)}$ refers to the diagonal element of inertia matrix A corresponding to the generalized acceleration \dot{q}_1 , J_{com} is the Jacobian of the CoM.

3) *Receding Horizon Control*: Dividing (13) by ML (where we defined $L = \sqrt{X_{com}^2 + Z_{com}^2}$, $X_{com} = L \sin\theta$ and $Z_{com} = L \cos\theta$) we get:

$$(\alpha + \cos\theta)\ddot{x} + (\beta + \cos\theta)\ddot{\theta} = g \sin\theta - R \sin\theta\dot{\theta}^2 \quad (14)$$

where $\alpha = \frac{R}{ML}(M + m + \frac{I_w}{R^2})$ and $\beta = \frac{I}{ML}$. Next by defining $X = [\theta \ \dot{\theta} \ x \ \dot{x}]^\top$ and $u = \ddot{\theta}$, we can rewrite the dynamics as:

$$\dot{X} = \begin{bmatrix} \dot{\theta} \\ u \\ \dot{x} \\ g \sin\theta - R \sin\theta\dot{\theta}^2 - (\beta + \cos\theta)u \\ \alpha + \cos\theta \end{bmatrix} = f_c(X, u). \quad (15)$$

For allowing the high-level controller to plan spin motion, $\dot{\psi}$ equation derived for the single link case from (3) will be added to the above. In that case, the control input is two-dimensional *i.e.*, $u = [\ddot{\theta}_{ref} \ \tau_0]^\top$, and we add four more

states ψ , $\dot{\psi}$, X_0 and Y_0 to the state vector, where (X_0, Y_0) is the location in the ground plane of the midpoint between the wheels wrt an inertial frame. ψ and $\dot{\psi}$ will evolve according to the $\ddot{\psi}$ equation of the WIPM. While X_0 and Y_0 will evolve according to $\dot{X}_0 = \dot{x} \cos\psi$ and $\dot{Y}_0 = \dot{x} \sin\psi$.

Assuming small time-step Δt the discretized dynamics of the system can be approximated as

$$X_{i+1} = X_i + \Delta t f_c(X_i, u_i) = f(X_i, u_i).$$

Given a desired goal position $X^{ref} = [\theta^{ref} \ \dot{\theta}^{ref} \ x^{ref} \ \dot{x}^{ref}]^\top = [0 \ 0 \ x^{ref} \ 0]^\top$ for a given final time t_f , define a one step cost

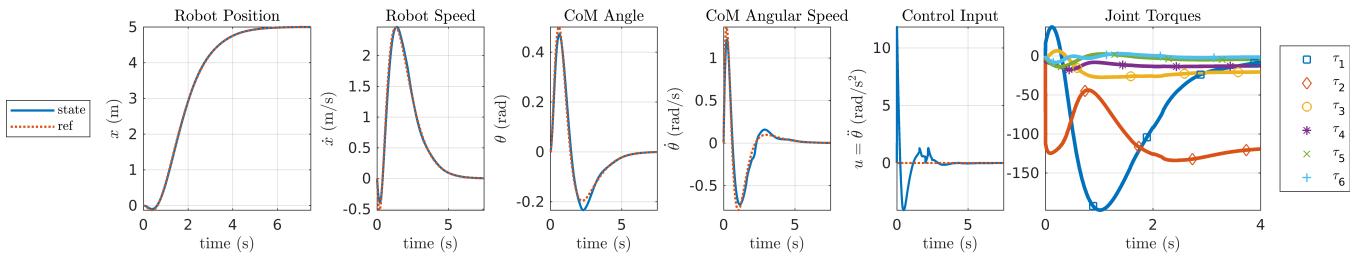
$$L_i^{ddp} = (X_i - X^{ref})^\top G_{ddp}(X_i - X^{ref}) + g_{ddp}u_i^2,$$

where G_{ddp} and g_{ddp} are weights penalizing state deviation and control effort respectively. Then DDP can be used to generate a reference trajectory. This trajectory will be generated over the full horizon t_f (or $N_{traj} = t_f/\Delta t$ steps), using the simplified model parameters $\lambda(q)$ at the initial state $q(0)$. Once we have the trajectory, $X^{traj}(i) = [\theta^{traj}(i) \ \dot{\theta}^{traj}(i) \ x^{traj}(i) \ \dot{x}^{traj}(i)]^\top$, it will be used as a time-varying reference for a receding horizon controller (or model predictive controller MPC) with a smaller horizon t_H (or steps $N = t_H/\Delta t$) to generate closed loop control. Here the parameters of the simplified model $\lambda(q)$ will be updated at every time-step using the current state of the full robot. Define the one-step cost $L_i^{mpc} = (X_i - X_i^{traj})^\top G_{mpc}(X_i - X_i^{traj}) + g_{mpc}u_i^2$, the MPC scheme will generate a control sequence for the horizon t_H by minimizing the cost over the horizon t_H . The first step of this control trajectory ($\dot{\theta}^{ref}$) will be used as reference target for the low-level controller. The low-level controller will determine torques of all joints by minimizing a single step cost using quadratic programming as described earlier. This enables full-body participation in locomotion along with performing other manipulation tasks, such as end-effector orientation or gaze control.

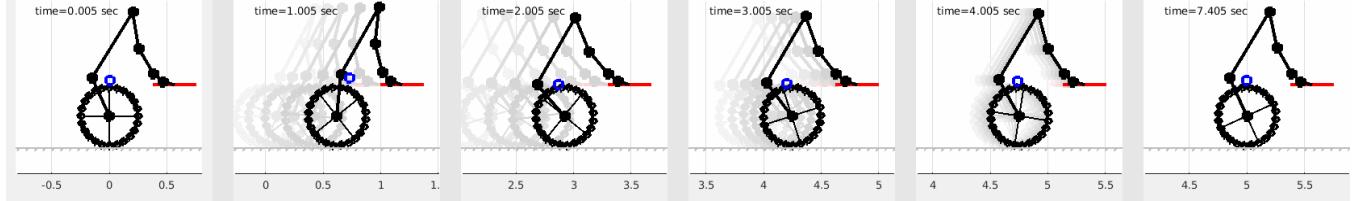
IV. RESULTS

We applied the presented approach on a simulation of 19-DOF robot (shown in Fig 1) with the objective of moving to a desired goal location on the ground while carrying a cup on a tray. The approach successfully manages to reach the goal location without letting the cup fall. We also make a comparison with the case of a traditional control [37], *i.e.*, without a unified control, to demonstrate its failure to prevent the cup from falling. The 3D simulation is uploaded as a video submission supplementing this paper. For the purpose of illustration, we provide here the details of applying the presented approach on a scaled down 7-DOF version of the same.

For trajectory generation using DDP, typically, a small penalty is assigned for each step and a large penalty is assigned for the terminal step. We have used a similar scheme. In our experiments, it turns out that for robots with large masses, a higher weight needs to be assigned



(a) Simplified model reference and state trajectories (first 5 plots). The last plot shows the resulting joint torques for $t \in (0, 4)$ seconds.



(b) Snapshots of the full body at 6 different instances during execution. For each snapshot, body pose at five recent instances is also shown using shaded lines, to visualize speeds. Blue circle represents body CoM, and the red line represents an object attached to the last link at a fixed orientation. We see that the whole body is participating in manipulating the CoM to fulfill position objectives, while maintaining the orientation and position of the object attached to the end-effector.

Fig. 3

to the pendulum angle $\theta(t)$. This ensures that the trajectory generated remains well within the stable region thus ensuring that the closed loop control (MPC) remains stable during execution. Note that MPC will not follow the trajectory generated by DDP exactly owing to the disturbances caused by full robot dynamics that were ignored in DDP. This means that θ may overshoot beyond the reference trajectory generated by DDP during execution. If θ was barely held within stable region, this may lead the full model to go unstable. For MPC, a horizon of 1 sec is used for each optimization step. A higher step cost is assigned to positions ($x(t)$ and $\theta(t)$) compared to speeds ($\dot{x}(t)$ and $\dot{\theta}(t)$). We have also used terminal weights for the MPC scheme, as they provide better tracking and stability performance. Terminal weights assign a high cost to the deviation of state at the end of the horizon at each control iteration. Final time $t_f = 20$ sec is used in the DDP to generate the results shown here. However the task was completed in 7.5 sec.

Finally, for the low-level controller, we have the following P and b matrices

$$P = \begin{bmatrix} w_\theta J_\theta \\ w_{r_{ee}} J_{r_{ee}} \\ w_{\phi_{ee}} J_{\phi_{ee}} \\ w_{reg} \mathcal{J}_6 \end{bmatrix} \quad b = \begin{bmatrix} -w_\theta (J_\theta \dot{q} - \ddot{\theta}^{ref}) \\ -w_{r_{ee}} (J_{r_{ee}} \dot{q} - \ddot{r}_{ee}^{ref}) \\ -w_{\phi_{ee}} (J_{\phi_{ee}} \dot{q} - \ddot{\phi}_{ee}^{ref}) \\ w_{reg} \ddot{q}_{reg} \end{bmatrix} \quad (16)$$

For each task $x_t \in \{\theta, r_{ee}, \phi_{ee}\}$ the reference acceleration \ddot{x}_t^{ref} appears in the definition of b and is defined as

$$\ddot{x}_t^{ref} = \ddot{x}^d - K_p(x_t - x_t^d) - K_v(\dot{x}_t - \dot{x}_t^d)$$

Desired position and orientation of the end-effector (r_{ee}^d and ϕ_{ee}^d) were set to their values at initial time. And the desired speeds and accelerations were set to zero. For pendulum angle θ , the desired position and speed (θ^d and $\dot{\theta}^d$) come

from the trajectory generated by DDP, while the desired acceleration $\ddot{\theta}^d$ is the control input determined by the MPC iteration. These reference values from DDP/MPC are used for the remaining sampling period (0.01 sec). During this period variable adaptive time-step simulation is used with low-level control QPs to simulate the behavior of the full robot.

Figure 3a shows the reference trajectory determined by DDP on the simplified model, and the state of the simplified model that results by applying the MPC control values on the full model. We see that the state follows the reference trajectory very closely with disturbances occurring during fast transitions owing to the disturbances caused by unmodeled full body dynamics. Also the last plot to the right is the plot of all joint torques. It is clear that the entire body is participating in performing the three tasks.

V. CONCLUSION

We presented a hierarchical approach to achieve whole body control of WIP humanoids where a fast QP-based low-level controller generates body torques based on operational space objectives for manipulation and a closed-loop CoM motion policy generated by a high-level controller. The latter utilizes MPC on a simplified model over a longer horizon. Time-varying reference for MPC is generated by DDP over a much larger horizon. Equations of motion for a typical WIP humanoid helped with inverse dynamics QPs, and for showing that body weight torque dominates wheel dynamics motivating the approximation used for model predictions in the high-level controller. We demonstrated the approach on a 19-DOF robot and presented a detailed account of a scaled down 7-DOF version of the same.

REFERENCES

- [1] K. D. (2001) The segway personal transporter (pt), the first self-balancing, zero emissions personal transportation vehicle. [Online]. Available: https://en.wikipedia.org/wiki/Segway_PT
- [2] J. Solis, R. Nakadate, Y. Yoshimura, Y. Hama, and A. Takanishi, "Development of the two-wheeled inverted pendulum type mobile robot wv-2r for educational purposes," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 2347–2352.
- [3] C.-C. Tsai, H.-C. Huang, and S.-C. Lin, "Adaptive neural network control of a self-balancing two-wheeled scooter," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 4, pp. 1420–1428, 2010.
- [4] Z. Li and C. Yang, "Neural-adaptive output feedback control of a class of transportation vehicles based on wheeled inverted pendulum models," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 6, pp. 1583–1591, 2012.
- [5] H. Hata and T. Takimoto, "Development of the portable two-wheeled inverted pendulum type personal vehicle," in *Control, Automation and Systems (ICCAS), 2014 14th International Conference on*. IEEE, 2014, pp. 1610–1613.
- [6] C.-N. Huang, "The development of self-balancing controller for one-wheeled vehicles," *Engineering*, vol. 2, no. 04, p. 212, 2010.
- [7] P. Petrov and M. Parent, "Dynamic modeling and adaptive motion control of a two-wheeled self-balancing vehicle for personal transport," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, 2010, pp. 1013–1018.
- [8] L. Vermeiren, A. Dequidt, T. M. Guerra, H. Rago-Tirmant, and M. Parent, "Modeling, control and experimental verification on a two-wheeled vehicle with free inclination: An urban transportation system," *Control Engineering Practice*, vol. 19, no. 7, pp. 744–756, 2011.
- [9] Y. Takahashi, S. Ogawa, and S. Machida, "Front wheel raising and inverse pendulum control of power assist wheel chair robot," in *Industrial Electronics Society, 1999. IECON'99 Proceedings. The 25th Annual Conference of the IEEE*, vol. 2. IEEE, 1999, pp. 668–673.
- [10] Y. Takahashi, T. Takagaki, J. Kishi, and Y. Ishii, "Back and forward moving scheme of front wheel raising for inverse pendulum control wheel chair robot," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 4. IEEE, 2001, pp. 3189–3194.
- [11] Y. Takahashi, N. Ishikawa, and T. Hagiwara, "Soft raising and lowering of front wheels for inverse pendulum control wheel chair robot," in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, vol. 4. IEEE, 2003, pp. 3618–3623.
- [12] Y. Takahashi and M. Kohda, "Human riding experiments on soft front wheel raising of robotic wheelchair with inverse pendulum control," in *Industrial Technology, 2005. ICIT 2005. IEEE International Conference on*. IEEE, 2005, pp. 266–271.
- [13] S. Jeong and T. Takahashi, "Wheeled inverted pendulum type assistant robot: inverted mobile, standing, and sitting motions," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. IEEE, 2007, pp. 1932–1937.
- [14] K. Sasaki and T. Murakami, "Pushing operation by two-wheel inverted mobile manipulator," in *2008 10th IEEE International Workshop on Advanced Motion Control*. IEEE, 2008, pp. 33–37.
- [15] T. Murakami and K. Sasaki, "A motion control for two-wheel inverted pendulum type of mobile manipulator," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 4, no. 2, pp. 192–198, 2009.
- [16] C. Acar and T. Murakami, "Multi-task control for dynamically balanced two-wheeled mobile manipulator through task-priority," in *2011 IEEE International Symposium on Industrial Electronics*. IEEE, 2011, pp. 2195–2200.
- [17] S. R. Kuindersma, E. Hannigan, D. Ruiken, and R. A. Grupen, "Dexterous mobility with the ubot-5 mobile manipulator," in *Advanced Robotics, 2009. ICAR 2009. International Conference on*. IEEE, 2009, pp. 1–7.
- [18] M. Stilman, J. Olson, and W. Gloss, "Golem krang: Dynamically stable humanoid robot for mobile manipulation," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 3304–3309.
- [19] T. Feng, T. Liu, X. Wang, Z. Xu, M. Zhang, and S.-c. Han, "Modeling and implementation of two-wheel self-balancing robot equipped with supporting arms," in *Industrial Electronics and Applications (ICIEA), 2011 6th IEEE Conference on*. IEEE, 2011, pp. 713–718.
- [20] H. Fukushima, K. Muro, and F. Matsuno, "Sliding-mode control for transformation to an inverted pendulum mode of a mobile robot with wheel-arms," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 7, pp. 4257–4266, 2015.
- [21] B. Dynamics. (2017) Boston dynamics officially unveils its wheel-leg robot: "best of both worlds". [Online]. Available: <http://spectrum.ieee.org/automaton/robotics/humanoids/boston-dynamics-handle-robot>
- [22] F. Grasser, A. D'arrigo, S. Colombi, and A. C. Rufer, "Joe: a mobile, inverted pendulum," *IEEE Transactions on industrial electronics*, vol. 49, no. 1, pp. 107–114, 2002.
- [23] J. Li, X. Gao, Q. Huang, Q. Du, and X. Duan, "Mechanical design and dynamic modeling of a two-wheeled inverted pendulum mobile robot," in *2007 IEEE International Conference on Automation and Logistics*. IEEE, 2007, pp. 1614–1619.
- [24] V. Coelho, S. Liew, K. Stol, and G. Liu, "Development of a mobile two-wheel balancing platform for autonomous applications," in *Mechatronics and Machine Vision in Practice, 2008. M2VIP 2008. 15th International Conference on*. IEEE, 2008, pp. 575–580.
- [25] K. Pathak, J. Franch, and S. K. Agrawal, "Velocity and position control of a wheeled inverted pendulum by partial feedback linearization," *Robotics, IEEE Transactions on*, vol. 21, no. 3, pp. 505–513, 2005.
- [26] D. Angeli, "Almost global stabilization of the inverted pendulum via continuous state feedback," *Automatica*, vol. 37, no. 7, pp. 1103–1108, 2001.
- [27] H. Marzi, "Multi-input fuzzy control of an inverted pendulum using an armature controlled dc motor," *Robotica*, vol. 23, no. 06, pp. 785–788, 2005.
- [28] W. Qingcheng and F. Jian, "Fuzzy immune pd algorithm applied in the self-balancing two-wheeled robot," in *Future Generation Communication and Networking (FGCN), 2014 8th International Conference on*. IEEE, 2014, pp. 112–115.
- [29] T.-J. Ren, T.-C. Chen, and C.-J. Chen, "Motion control for a two-wheeled vehicle using a self-tuning pid controller," *Control Engineering Practice*, vol. 16, no. 3, pp. 365–375, 2008.
- [30] S. Jung and S. S. Kim, "Control experiment of a wheel-driven mobile inverted pendulum using neural network," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 2, pp. 297–303, 2008.
- [31] J. Wu and S. Jia, "Ts adaptive neural network fuzzy control applied in two-wheeled self-balancing robot," in *Strategic Technology (IFOST), 2011 6th International Forum on*, vol. 2. IEEE, 2011, pp. 1023–1026.
- [32] J. Wu and W. Zhang, "Design of fuzzy logic controller for two-wheeled self-balancing robot," in *Strategic Technology (IFOST), 2011 6th International Forum on*, vol. 2. IEEE, 2011, pp. 1266–1270.
- [33] L. B. Prasad, B. Tyagi, and H. O. Gupta, "Optimal control of nonlinear inverted pendulum dynamical system with disturbance input using pid controller & lqr," in *Control System, Computing and Engineering (ICCSCE), 2011 IEEE International Conference on*. IEEE, 2011, pp. 540–545.
- [34] S. Ahmad, M. O. Tokhi, and S. F. Toha, "Genetic algorithm optimisation for fuzzy control of wheelchair lifting and balancing," in *Computer Modeling and Simulation, 2009. EMS'09. Third UKSim European Symposium on*. IEEE, 2009, pp. 97–101.
- [35] K. M. GOHER and M. Tokhi, "Ga-optimised steering and position control of a two-wheeled vehicle with an extended rod-a simulation study," in *The 12th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines, Istanbul, Turkey*, 2009, pp. 66–74.
- [36] K. M. Goher and M. Tokhi, "Genetic algorithm based modeling and control of a two wheeled vehicle with an extended rod, a lagrangian based dynamic approach," in *Cybernetic Intelligent Systems (CIS), 2010 IEEE 9th International Conference on*. IEEE, 2010, pp. 1–6.
- [37] M. Stilman, M. Zafar, C. Erdogan, P. Hou, S. Reynolds-Haertle, and G. Tracy, "Robots using environment objects as tools the mac-gyverparadigm for mobile manipulation," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2568–2568.
- [38] S. Feng, "Online hierarchical optimization for humanoid control," 2016.
- [39] J. Akesson, A. Blomdell, and R. Braun, "Design and control of yaipan inverted pendulum on two wheels robot," in *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*. IEEE, 2006, pp. 2178–2183.

- [40] Y. Ha and S. Yuta, "Trajectory tracking control for navigation of self-contained mobile inverse pendulum," in *Intelligent Robots and Systems' 94.'Advanced Robotic Systems and the Real World', IROS'94. Proceedings of the IEEE/RSJ/GI International Conference on*, vol. 3. IEEE, 1994, pp. 1875–1882.
- [41] H. Jian-hai, Z. Shu-shang, L. Ji-shun, and L. Hang, "Research on developed parallel two-wheeled robot and its control system," in *2008 IEEE International Conference on Automation and Logistics*. IEEE, 2008, pp. 2471–2475.
- [42] Y. Kim, S.-H. Lee, and D. H. Kim, "Dynamic equations of a wheeled inverted pendulum with changing its center of gravity," in *Control, Automation and Systems (ICCAS), 2011 11th International Conference on*. IEEE, 2011, pp. 853–854.
- [43] J. Lien, L. Tu, W. Ross, and C. Burvill, "Implementation issues for an inexpensive inverted-pendulum mobile robot," in *2006 International Conference on Information and Automation*. IEEE, 2006, pp. 372–377.
- [44] T. Nomura, Y. Kitsuka, H. Suemitsu, and T. Matsuo, "Adaptive backstepping control for a two-wheeled autonomous robot," in *ICCAS-SICE, 2009*. IEEE, 2009, pp. 4687–4692.
- [45] R. C. Ooi, "Balancing a two-wheeled autonomous robot," *University of Western Australia*, vol. 3, 2003.
- [46] J.-S. Hu and M.-C. Tsai, "Design of robust stabilization and fault diagnosis for an auto-balancing two-wheeled cart," *Advanced Robotics*, vol. 22, no. 2-3, pp. 319–338, 2008.
- [47] T. Takei, R. Imamura *et al.*, "Baggage transportation and navigation by a wheeled inverted pendulum mobile robot," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 3985–3994, 2009.
- [48] M.-C. Tsai and J.-S. Hu, "Pilot control of an auto-balancing two-wheeled cart," *Advanced Robotics*, vol. 21, no. 7, pp. 817–827, 2007.
- [49] J. Wu, Y. Liang, and Z. Wang, "A robust control method of two-wheeled self-balancing robot," in *Strategic Technology (IFOST), 2011 6th International Forum on*, vol. 2. IEEE, 2011, pp. 1031–1035.
- [50] J. Lee, M. X. Grey, S. Ha, T. Kunz, S. Jain, Y. Ye, S. S. Srinivasa, M. Stilman, and C. K. Liu, "Dart: Dynamic animation and robotics toolkit," *The Journal of Open Source Software*, vol. 3, no. 22, p. 500, 2018.
- [51] R. Featherstone, "Robot dynamics algorithms," 1984.
- [52] J. H. Ginsberg, *Advanced engineering dynamics*. Cambridge University Press, 1998.
- [53] Z.-Q. Guo, J.-X. Xu, and T. H. Lee, "Design and implementation of a new sliding mode controller on an underactuated wheeled inverted pendulum," *Journal of the Franklin Institute*, vol. 351, no. 4, pp. 2261–2282, 2014.
- [54] M. Zafar and H. I. Christensen, "Whole body control of a wheeled inverted pendulum humanoid," in *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*. IEEE, 2016, pp. 89–94.
- [55] D. Jacobson and D. Mayne, *Differential dynamic programming*, ser. Modern analytic and computational methods in science and mathematics. American Elsevier Pub. Co., 1970. [Online]. Available: <https://books.google.com/books?id=tA-oAAAAIAAJ>
- [56] Y. Tassa, T. Erez, and E. Todorov, "Synthesis and stabilization of complex behaviors through online trajectory optimization," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 4906–4913.