

Robust Humanoid Contact Planning with Learned Zero- and One-Step Capturability Prediction

Yu-Chi Lin¹, Ludovic Righetti^{2,3}, and Dmitry Berenson¹

Abstract—Humanoid robots maintain balance and navigate by controlling the contact wrenches applied to the environment. While it is possible to plan dynamically-feasible motion that applies appropriate wrenches using existing methods, a humanoid may also be affected by external disturbances. Existing systems typically rely on controllers to reactively recover from disturbances. However, such controllers may fail when the robot cannot reach contacts capable of rejecting a given disturbance. In this paper, we propose a search-based footstep planner which aims to maximize the probability of the robot successfully reaching the goal without falling as a result of a disturbance. The planner considers not only the poses of the planned contact sequence, but also alternative contacts near the planned contact sequence that can be used to recover from external disturbances. Although this additional consideration significantly increases the computation load, we train neural networks to efficiently predict multi-contact zero-step and one-step capturability, which allows the planner to generate robust contact sequences efficiently. Our results show that our approach generates footstep sequences that are more robust to external disturbances than a conventional footstep planner in four challenging scenarios.

Index Terms—Motion and Path Planning; Humanoid and Bipedal Locomotion; Deep Learning in Robotics and Automation

I. INTRODUCTION

Algorithms efficiently computing contact sequences to traverse complex terrains are a fundamental building block for multi-contact behaviors of legged robots, in particular humanoids. In order to reduce computational complexity, most contact planners generate contact sequences considering solely quasi-static constraints [1–5]. However, a static stability criterion significantly decreases the set of possible contact transitions, which quickly leads to planning failure when attempting to traverse complex environments. More recently, efficient planners using more general dynamic feasibility constraints have also been proposed [6–8]. Nevertheless, all these approaches assume fixed, deterministic environments and do not consider the robustness of contact sequences to potential environmental disturbances. In Figure 1, we show an example where the robot walks over rubble. There is a wall in the environment, and the robot can use palm contacts to capture itself against potential disturbances. However, without considering this information in the planner, a conventional contact planner could take the shortest feasible path which

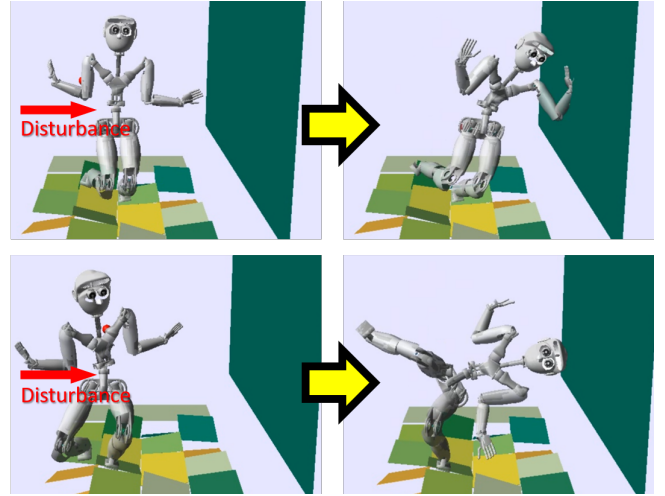


Fig. 1. The robot walks over a rubble, and is impacted by a disturbance. Top: The robot walks close to the wall, and capture itself using a palm contact on the wall. Bottom: The robot cannot reach the wall, and falls down under the disturbance.

does not have access to the wall, and may cause the robot to fall down when a disturbance occurs.

In this paper, we propose a computationally efficient footstep planner that explicitly takes into account disturbances to increase motion robustness. In particular, we consider zero-step and one-step capture motions using either foot or palm contacts. Testing the existence of capture motions in multi-contact scenarios necessitates the solution to a kino-dynamic optimal control problem [9], [10]. However, it is prohibitively long to directly solve such problem in a footstep planner, as every candidate contact transition requires such a test. Instead, we propose to train neural networks to predict the existence of a dynamically feasible capture motion using data generated offline with a kino-dynamic optimizer. The networks predict both zero-step and one-step capturability for a full-body dynamic model using both foot and palm capture motions. We then query these networks in the footstep planning loop to inform the Anytime Non-parametric A*(ANA*) planner [11] about which footstep transitions are most robust to disturbances by measuring how many sampled contact poses can reject the disturbances. To the best of our knowledge, this work is the first footstep planner to use a learned model that predicts robot capturability under disturbances to produce more robust footstep sequences.

Our experiments first show that our neural networks achieve high accuracy in predicting robot capturability. We then compare our planning approach to a conventional distance-based footstep planner. Our results show that our approach generates footstep sequences that are more robust to external disturbances than the conventional method in four challenging scenarios.

Manuscript received: September, 10th, 2019; Revised December, 12th, 2019; Accepted January, 21st, 2020

This paper was recommended for publication by Editor Nancy Amato upon evaluation of the Associate Editor and Reviewers' comments. This work was supported in part by the Office of Naval Research under grant N000141712050

¹University of Michigan, Ann Arbor, MI, USA linyuchi@umich.edu, berenson@eecs.umich.edu

²Max Planck Institute for Intelligent Systems, Tuebingen, Germany

³New York University, New York, NY, USA ludovic.righetti@nyu.edu

Digital Object Identifier (DOI): see top of this page.

II. RELATED WORK

Humanoid footstep planning has been broadly studied [12–18]. Conventional approaches plan a footstep sequence on a flat or piecewise-flat ground to avoid obstacles. To increase planning efficiency, these planners define a contact transition model which assumes all motions are dynamically feasible throughout the operating environment, and plan footstep sequences without explicitly checking their dynamic feasibility. Therefore, it is difficult for this kind of approach to generalize across environments.

More recent works verify dynamic feasibility by approximating it with quasi-static balance [1–4], [19]. These approaches are able to consider more diverse actions, including multi-contact motion. However, the quasi-static balance criterion is too conservative to consider dynamic motions.

To address the over-conservative nature of the quasi-static balance criterion, there are works which combine contact planning with dynamics optimization by solving a mixed integer convex program [8], [20], [21]. This approach produces the global optimal solution, but does not generalize well to the complexity of the scene and the path length. To increase efficiency in checking dynamic feasibility, [6] proposes an approach by conservatively reformulating the problem as a linear program. Our previous work [7] proposes a graph-search based contact planner which uses neural networks to quickly predict the result of a dynamics optimizer. In this work, we further consider the robustness of the planned contact sequence under external disturbances.

Capturability analysis of the linear inverted pendulum (LIP) model was first proposed by [22]. Since then, it has been widely used to determine footstep placement in planning and control of robot dynamic walking [23–28]. There are also works performing capturability analysis for the more complex variable-height inverted pendulum (VHIP) model to account for the height changes of the CoM. [29–31] address the balance control of humanoid robot using VHIP model for planar motions. [32] further extends it to consider 3D movements, and develops an analytical tool to determine capturability in the VHIP model. [9] proposes an efficient analytical tool to compute zero-step capturability for multi-contact configuration using a centroidal dynamics model. However, it has strong assumptions on using zero angular momentum, and cannot generalize to use additional steps.

III. PROBLEM STATEMENT

We focus on the problem of planning humanoid footstep sequences considering the effect of external disturbances. Given an environment specified as a set of polygonal surfaces, an initial stance (set of poses of contacting end-effectors), a goal region, and a distribution of potential disturbances in the environment, we aim to output a dynamically feasible footstep sequence to move the robot from the initial stance to the goal region. In the planning, we consider not only where the robot can create contacts to achieve dynamically feasible motions, but also how well the robot can capture itself with existing and nearby contact locations, using **both feet and hands**, to reject disturbances sampled from the distribution of potential disturbances. While it is important and desirable to generate those capture motions in real time, it is still an

open problem and beyond the scope of this work. Our goal is to find a footstep sequence that maximizes the probability of the robot reaching the goal successfully without falling as a result of a disturbance. Notice that only feet are used in locomotion, but both feet and hands are available for rejecting potential disturbances. We assume that the friction coefficient is given, as well as a fixed timing for each contact transition. In this work, we consider both zero-step and one-step capture motions.

IV. ITERATIVE KINO-DYNAMIC OPTIMIZATION

In order to decide whether a capturing motion exists for the full robot model, we use the kino-dynamic optimization method described in [33]. Given a sequence of collision-free contact poses, the method decomposes the problem of optimizing dynamically-consistent whole-body motions and contact forces into 1) a dynamic optimization problem based on the centroidal dynamics [34] and 2) a kinematic optimization problem for the full-body motions. The algorithm computes the solution of each problems iteratively until both parts reach consensus over the center of mass \mathbf{r} , linear \mathbf{l} and angular momentum \mathbf{k} trajectories, leading to a locally optimal solution of the original problem.

In this work, we use the algorithm proposed in [35] with fixed-time to efficiently compute a solution for the dynamic optimization problem. The centroidal dynamics expressed at the robot CoM is given by

$$\begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{l}} \\ \dot{\mathbf{k}} \end{bmatrix} = \begin{bmatrix} \frac{1}{M}\mathbf{l} \\ M\mathbf{g} + \sum \mathbf{f}_e \\ \sum (T_e(\mathbf{z}_e) - \mathbf{r}) \times \mathbf{f}_e + \tau_e \end{bmatrix} \quad (1)$$

M is the robot mass. \mathbf{z}_e is the center of pressure (CoP) of each contact in the contact frame. \mathbf{f}_e and τ_e are the contact force and torque at the CoP of each end-effector and finally, T_e is a coordinate transform in the CoM frame. In addition to Eq. (1), contact forces need to be inside friction cones, and CoPs inside the support regions of each contact, to prevent the contact from sliding and tilting.

To compute a dynamically robust motion we follow [35] to minimize the weighted sum of the square norm of \mathbf{l} , $\dot{\mathbf{l}}$, \mathbf{k} , $\dot{\mathbf{k}}$, \mathbf{f}_e , and τ_e . Lower values of \mathbf{l} and $\dot{\mathbf{l}}$ help improve dynamic stability [36]. Reducing \mathbf{k} and $\dot{\mathbf{k}}$ help the robot perform more natural motion [37]. The \mathbf{f}_e and τ_e terms encourage a more even distribution of forces and torques over all the contacts, which increases the controllability of the robot. The dynamic optimizer is run before the kinematic optimizer. After the first iteration, torque limits are included in the dynamic optimizer by using the kinematic solution to find an approximation of the torque changes during the centroidal dynamics optimization. To simplify the problem, in this work, collision avoidance is not considered in the optimization. In future works, we would like to incorporate the collision constraints using methods in [38].

A contact transition is considered capturable if the algorithm converges to consensus to a solution that satisfies all constraints after a maximum number of iterations, where we set constraints on the linear and angular momenta at the end of the movement to zero to ensure the robot will come to a stop.

V. MODELING EXTERNAL DISTURBANCES

We model an external disturbance as an **instant** change in linear centroidal momentum. Therefore, an external disturbance δ is a 3D vector: $\delta \in \mathbb{R}^3$. We assume there is a known probability distribution of potential disturbances in each location $\mathbf{x} \in \mathbb{R}^3$ in the environment and the distribution is fixed during planning and execution time. To facilitate capturability checking, we discretize the distribution by sampling a set of representative disturbances from the distribution, and the probability of each disturbance sample is the total probability integrated over the Voronoi cell of the disturbance sample. Let $D(\mathbf{x})$ be the set of all representative disturbances. We assume that for any short period of time T , there will only be one disturbance, so we have

$$\sum_{i=1}^{N_D(\mathbf{x})} P(\delta_i, T) = 1, \quad D(\mathbf{x}) = \{\delta_i | i = 1, 2, \dots, N_D(\mathbf{x})\} \quad (2)$$

where $P(\delta_i, T)$ is the probability that δ_i happens once within time duration T , and $N_D(\mathbf{x})$ is the number of disturbance samples in $D(\mathbf{x})$.

VI. EVALUATION OF CAPTURABILITY

To evaluate capturability, we adopt the approach of iterative kino-dynamic optimization described in Section IV. Since we model the disturbance as an instant change in linear momentum, we use the post-disturbance centroidal dynamics state $[\mathbf{r}_0, \mathbf{l}_0, \mathbf{k}_0]^T$, the centroidal dynamics state immediately after the disturbance δ , as the initial state of the iterative kino-dynamic optimization, and define it as $[\mathbf{r}_0, \mathbf{l}_0, \mathbf{k}_0]^T = [\mathbf{r}_b, \mathbf{l}_b + \delta, \mathbf{k}_b]^T$, where $[\mathbf{r}_b, \mathbf{l}_b, \mathbf{k}_b]^T$ is the centroidal dynamics state before disturbance.

In this work, we consider two kinds of capture motions: zero-step capture (capturing without making new contacts), and one-step capture (capturing by making one new contact). For zero-step capture, the initial condition of the optimization includes $[\mathbf{r}_0, \mathbf{l}_0, \mathbf{k}_0]^T$, and existing contact poses. For one-step capture, in addition to the above initial conditions, we also specify a target contact pose for one of the free end-effectors.

To determine capturability, we first optimize the initial kinematic states $[\mathbf{q}_0, \dot{\mathbf{q}}_0]$ to track $[\mathbf{r}_0, \mathbf{l}_0, \mathbf{k}_0]^T$ and the existing stance S (set of contacting end-effectors poses), and then run kino-dynamic optimization for three iterations. If a kino-dynamically feasible solution can be found such that the linear and angular momentum converge to zero at the end of the motion, then the robot is capturable under the specified initial conditions: $(\mathbf{r}_0, \mathbf{l}_0, \mathbf{k}_0, S)$. For the one-step capture case, we try three different durations for the robot to move the end-effector to make contact: 0.2, 0.4 and 0.6 seconds. If any duration is feasible, then the robot is capturable given the initial conditions. The evaluation for each contact pose takes from the order of 100 ms to 1 s depending on the difficulty of the situation and the number of iterations attempted. Although it is prohibitively long to be included in a planning loop, we can collect the result offline, and fit it with an computationally efficient model.

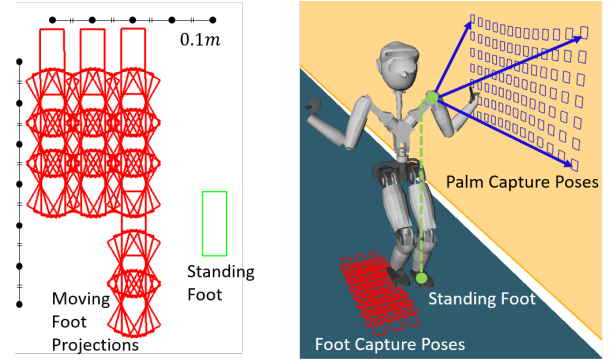


Fig. 2. (a) Left: Foot contact transition model in searching contact sequence, (b) Right: Possible foot and palm contact projections for one-step capture motion given the standing foot pose. The projections are shown on flat surfaces as an illustrative example. When generating training data we sample contact poses with random tilt angles.

Index	Capture Motion Type	Capture Motion	Input Dim.
0	Zero-Step Capture	Maintain one foot contact	12
1	One-Step Capture	Make the other foot contact	18
2		Make the same side palm contact	18
3		Make the opposite side palm contact	18

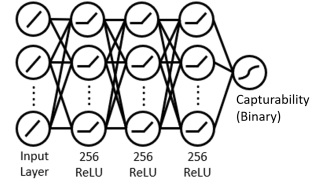


Fig. 3. Left: Capture motions considered in this work and their feature dimension. Every capture motion initially has one foot contact, and the side of the palm contacts is relative to the standing foot side. Right: The network structure to predict capturability. The learning rate is 5×10^{-5} and there are dropout layers between fully-connected layers with 0.1 dropout rate.

VII. LEARNING THE RESULT OF THE KINO-DYNAMIC OPTIMIZATION OF CAPTURE MOTIONS

For each contact transition evaluated in contact planning, the planner needs to decide if the robot can capture itself under a set of disturbances D , and for each disturbance $\delta_i \in D$, many potential contacts may be considered to capture the robot in one step. Therefore, it is computationally prohibitive to run the iterative kino-dynamic optimization in the planning loop. To reduce online computation, we train a set of neural network classifiers offline to determine capturability. Each neural network corresponds to a separate capture motion involving different contacts, as shown in Figure 3.

The classifiers predict whether the optimizer can find a kino-dynamically feasible solution to capture the robot given the initial conditions described in Section VI. Since angular momenta are generally low in walking motion [37], we assume $\mathbf{k}_0 = 0$ and do not include it as the input of the network to improve data efficiency. As shown in Figure 3, the classifiers take the initial standing foot pose, the capture contact pose, and $[\mathbf{r}_0, \mathbf{l}_0]^T$ as inputs, and have a 1D binary output, which represents whether the optimizer can find a kino-dynamically feasible solution to capture the robot. Because most humanoid robots have symmetric kinematic structures, we utilize this symmetry and define 4 kinds of capture motion, as shown in Figure 3. For zero-step capture cases, the involved contact poses are only the existing contact poses; for one-step capture cases, a new contact pose of a free end-effector is considered. Each contact pose is a \mathbb{R}^6 vector which consists of position and orientation in Tait-Bryan angles, convention X-Y-Z, in $[-\pi, \pi)$. To capture the spatial relationship of the orientation with angles near $\pm\pi$, we duplicate those samples with $\mp 2\pi$ in the training data.

To collect meaningful training data, we determine the sampling space based on the robot's reachability and the target application. If we randomly sample contact poses in a wide space, most samples are not feasible or not useful for our application. To address this issue, we first get a rough estimate of the robot's reachability with kinematic optimizers on a set of widely-sampled contact poses, and then reduce the sampling space by defining sampling intervals in each dimension of $SE(3)$ to focus more on the robot's reachable space and poses required by the application. Although the training data will need to be recollected if different robots or applications are considered, in this way, we can get a more balanced data set.

In this work, we collect data by sampling the initial standing foot contact pose with random tilt angles within $\pm 25^\circ$ from Z axis. \mathbf{r}_0 is randomly sampled relative to the foot pose based on the robot's reachability, and \mathbf{l}_0 is randomly sampled in the magnitude interval of $m[0, 1]\text{kg} \cdot \text{m/s}$, where m is the robot mass, and its orientation is randomly sampled within $\pm 45^\circ$ from the XY plane. For one-step capture cases, we sample capture contact poses using models shown in Figure 2. Each contact is projected with randomly selected depth and tilt angle to form a diverse set of initial conditions. Each sampled initial condition is supplied to the kino-dynamic optimizer described in Section VI to decide its label. A different neural network is trained to determine capturability for each type of capture motion, but we use the same network structure for all capture motions to simplify the implementation, as shown in Figure 3.

VIII. ANYTIME DISCRETE-SEARCH CONTACT PLANNER

We formulate the contact planning problem as a graph search problem. Each state s in the graph is represented by a set of: a stance $S(s)$, a CoM position $\mathbf{r}(s)$, and a linear momentum $\mathbf{l}(s)$. Each action is a foot contact transition, which means moving one foot to a new pose. Contact transitions are predefined as a discrete set of foot projections, shown in Figure 2(a), and we adopt the contact projection approach in [39].

For each contact transition $\varepsilon(s, s')$ from state s to state s' , the planner generates a new state with a stance which differs from the current stance by the moving contact pose. We assume there is a 0.4 second long swing phase followed by 0.6 second double support phase for each contact transition. We follow our previous work [7], given $S(s)$, $S(s')$, $\mathbf{r}(s)$ and $\mathbf{l}(s)$, we use neural networks to predict dynamic feasibility of the contact transition, and determine $\mathbf{r}(s')$ and $\mathbf{l}(s')$.

We solve the contact planning problem with Anytime Non-parametric A*(ANA*) algorithm [11]. ANA* is an anytime variation of the A* algorithm. It initially inflates the heuristic and determines which node to expand mainly by evaluating its heuristic. Once a solution is found, it then reduces the inflation of the heuristic, and improves the solution over time. In this way, a feasible solution can be generated quickly, and helps reduce the search space to find a better solution over time. The cost of each action connecting two states s and s' is defined as

$$\Delta g(s, s') = d(s, s') + w_s + w_{cap}c_{cap}(s, s') \quad (3)$$

where $d(s, s')$ is the euclidean XY distance between the mean foot positions of state s and s' , w_s is a fixed step cost, and c_{cap} is the capturability cost and w_{cap} is its corresponding weight.

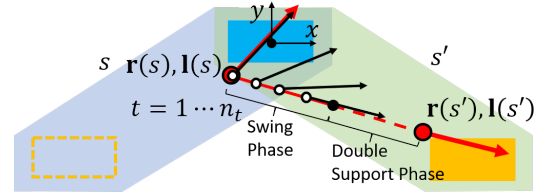


Fig. 4. Approximated CoM position and linear momentum used to check capturability in Swing Phase Discretization. Blue and yellow boxes represent standing and swing foot, respectively. In practice, we let $n_t = 4$ to represent 4 time steps in the swing phase: $0^+, 0.1, \dots, 0.3$ seconds from the start of the swing.

We aim to generate a contact sequence which maximizes the robot's success rate to reach the goal without falling due to disturbance. Therefore, c_{cap} should be determined by the probability that the robot can capture itself during the contact transition $\varepsilon(s, s')$ from s to s' given the probability distribution of the disturbances. We denote the capture probability as $P_{\text{success}}(\varepsilon(s, s'))$.

To determine $P_{\text{success}}(\varepsilon(s, s'))$, we consider two different approaches:

- Swing Phase Discretization: Considering n_t pairs of (\mathbf{r}, \mathbf{l}) from discretized time steps during the swing phase of contact transition from s to s' , as shown in Figure 4.
- Worst-case CoM Estimate: Considering only the (\mathbf{r}, \mathbf{l}) pair right after the robot breaks a contact to start the swing phase (approximated as $(\mathbf{r}(s), \mathbf{l}(s))$).

For Swing Phase Discretization, $P_{\text{success}}(\varepsilon(s, s'))$ is defined as

$$\prod_{t=1}^{n_t} \sum_{i=1}^{N_D(\mathbf{r}_t)} P_{\text{reject}}(\mathbf{r}_t, \mathbf{l}_t, S_{\text{swing}, \varepsilon(s, s')}, \delta_i) P\left(\delta_i, \frac{0.4}{n_t}\right) \quad (4)$$

$$\begin{cases} \mathbf{r}_t = \frac{n_t-t}{n_t-1}\mathbf{r}(s) + \frac{t-1}{n_t-1}\mathbf{r}_{\text{swing}, \varepsilon(s, s')} & t = 1, \dots, n_t \\ \mathbf{l}_t = \frac{n_t-t}{n_t-1}\mathbf{l}(s) + \frac{t-1}{n_t-1}\mathbf{l}_{\text{swing}, \varepsilon(s, s')} \end{cases}$$

where $P_{\text{reject}}(\mathbf{r}_t, \mathbf{l}_t, S_{\text{swing}, \varepsilon(s, s')}, \delta_i)$ means the probability of the robot rejecting disturbance $\delta_i \in D(\mathbf{r}(s))$ with centroidal dynamics state before disturbance $[\mathbf{r}_b, \mathbf{l}_b, \mathbf{k}_b]^T = [\mathbf{r}_t, \mathbf{l}_t, 0]^T$, and the robot's stance in swing phase $S_{\text{swing}, \varepsilon(s, s')} \cdot \mathbf{r}_{\text{swing}, \varepsilon(s, s')}$ and $\mathbf{l}_{\text{swing}, \varepsilon(s, s')}$ are \mathbf{r} and \mathbf{l} at the end of the swing phase, and they are set empirically to be $\mathbf{r}_{\text{swing}, \varepsilon(s, s')} = 0.4\mathbf{r}(s') + 0.6\mathbf{r}(s)$ and $\mathbf{l}_{\text{swing}, \varepsilon(s, s')} = \mathbf{l}(s')$, respectively. Although only time steps in swing phase are considered here, empirically we find that for each step cycle, the robot has similar performance to reject disturbances by reactive stepping in double support phase or one-step capture in swing phase. Therefore, to reduce the computation load, we sample only from the swing phase in planning.

For Worst-case CoM Estimate, $P_{\text{success}}(\varepsilon(s, s'))$ is defined as

$$\sum_{i=1}^{N_D(\mathbf{r}(s))} P_{\text{reject}}(\mathbf{r}(s), \mathbf{l}(s), S_{\text{swing}, \varepsilon(s, s')}, \delta_i) P(\delta_i, 0.4) \quad (5)$$

In this definition, $P_{\text{success}}(\varepsilon(s, s'))$ only depends on s and $S_{\text{swing}, \varepsilon(s, s')}$, so for all s' with the same $S_{\text{swing}, \varepsilon(s, s')}$, $P_{\text{success}}(\varepsilon(s, s'))$ is the same. Therefore, compared to Swing Phase Discretization, Worst-case CoM Estimate reduces the computation time significantly because it only considers one

centroidal dynamics state. During the contact transition, disturbances pushing toward $+y$ direction in standing foot frame are hard to capture with the swing foot because of the kinematic constraints. As seen in Figure 4, we observe that in dynamic walking, at the start of the swing phase, the robot has the highest $+y$ component of the linear momentum. Therefore, in Worst-case CoM Estimate, we sample the start of the swing phase of each $\varepsilon(s, s')$, and use it to determine $P_{\text{success}}(\varepsilon(s, s'))$.

A. Modelling disturbance rejection probability

Both definitions of $P_{\text{success}}(\varepsilon(s, s'))$ require the disturbance rejection probability $P_{\text{reject}}(\mathbf{r}, \mathbf{l}, S_{\text{swing}, \varepsilon}, \delta)$. For each $S_{\text{swing}, \varepsilon}$, we use the foot and palm projection model shown in Figure 2(b) to find all possible capture poses. We then query the neural networks with $\mathbf{r}, \mathbf{l}, S_{\text{swing}, \varepsilon}$ and each of those capture poses, and count the number of queries that output “capturable”, including the zero-step capture motion, denoted as n_c . Since the neural networks simplify the capturability check by abstracting the initial kinematics state to be a combination of a stance and a dynamics state, and assuming no initial angular momentum, we expect errors caused by these simplifications. Therefore, we would like to improve the planner robustness by favoring transitions $\varepsilon(s, s')$ which are predicted by the networks to be capturable with more capture poses (higher n_c for each disturbance). Therefore, we model $P_{\text{reject}}(\mathbf{r}, \mathbf{l}, S_{\text{swing}, \varepsilon}, \delta)$ as $1 - \exp(-\gamma n_c)$, where $\gamma \in \mathbb{R}^+$ is a user defined constant. This model captures the idea that the robot is more likely to reject the disturbance if more network queries with different capture poses determine the condition to be capturable.

B. Capturability Cost

For a path T_{cp} (a sequence of K contact transitions), the probability that the robot finishes the path without falling due to external disturbance is

$$P_{\text{success}}(T_{cp}) = \prod_{k=1}^K P_{\text{success}}(\varepsilon_k) \quad (6)$$

where ε_k is the k th contact transition in T_{cp} . Our goal is to maximize $P_{\text{success}}(T_{cp})$, which can be achieved by minimizing $\sum_{k=1}^K -\log(P_{\text{success}}(\varepsilon_k))$. Therefore, we define c_{cap} as

$$c_{cap}(s, s') = -\log(P_{\text{success}}(\varepsilon(s, s'))) \quad (7)$$

With this definition of c_{cap} , we can find a path with maximum success rate by minimizing the total capturability cost of the path, which is done by the ANA* algorithm. In practice, we set $w_{cap} \gg w_s, d(s, s')$ to let ANA* focus on maximizing $P_{\text{success}}(T_{cp})$.

C. Contact Planning Heuristic

To guide the search, we follow [7] and define the heuristic function by computing a policy for a simplified robot model moving on an SE(2) grid. The robot simplified model is a floating box. We first prune out every cell in the grid where there is no ground or there is collision between the box and the environment, and plan with Dijkstra’s algorithm from the goal cell using an 8-connected grid transition model. By doing so,

every cell connected to the goal cell will get a shortest distance $d_{\text{Dijkstra}}(s)$ to reach the goal and a policy which indicates the neighboring cell to go to. During contact planning, the planner queries this policy with the mean foot position on the XY plane, and the mean foot rotation about the Z axis to compute the heuristic.

$$h(s) = d_{\text{Dijkstra}}(s) + w_s \frac{d_{\text{Dijkstra}}(s)}{\Delta d_{max}} \quad (8)$$

where Δd_{max} is an overestimate of the maximum distance the mean foot pose can travel in one transition.

IX. EXPERIMENTS

We evaluate the performance of the proposed approaches in three test environments in simulation: a narrow, flat strip of ground, a field of rubble with an adjacent wall, and part of an oil platform, as shown in Figure 5. For each test, we allow 1 minute planning time, and set $w_s = 3$, $w_{cap} = 1000$, $\gamma = 0.1$ and the friction coefficient is 0.5. We compare the proposed approaches with the baseline approach which only considers moving distance and step number ($w_{cap} = 0$). For all test environments, we show the planned footstep sequences in Figure 5, and summarize the quantitative results in Figure 7.

Since small disturbances can be handled by the robot’s momentum controller, and do not require the planner to explicitly find capture motion to reject them, in the below experiment, we only consider the relatively rare but dangerous case that high disturbances D_{high} act on the robot. Unless otherwise stated, we set the probability of those high disturbances happening within every time step (0.1 second) as $P(D_{\text{high}}, 0.1) = 1\%$. To make the result easier to interpret, we let $P(\delta_i, 0.1)$, $\delta_i \in D_{\text{high}}$ evenly divide $P(D_{\text{high}}, 0.1)$.

To evaluate the planned contact sequence, we first get its corresponding kinematic trajectory using the iterative kinodynamic optimizer described in Section IV. Each trajectory is a discrete sequence of $\mathbf{q}, \dot{\mathbf{q}}$ with time steps of 0.1 second. For each time step t_j of the kinematic trajectory, including both swing and double support phases, we take the configuration as the initial kinematic state, and apply disturbances $\delta_i \in D_{\text{high}}(\mathbf{r}(t_j))$ one by one and check if the robot can capture itself using the approach described in Section VI. For each disturbance δ_i , we first check if the condition is zero-step capturable, if not, we then check if it is one-step capturable with any of the capture poses generated using contact projection shown in Figure 2. In double support phase, when testing one-step capturability, we allow the robot to break one existing contact, and make contact at a capture pose. With the capturability of the robot for each time step - disturbance pair, we finally compute the probability that the robot finishes the path without falling due to external disturbance $P_{\text{success}}(T_{cp})$ to evaluate the path quality.

We run the experiments on an Intel i7-8700K 3.7GHz CPU, and use an NVIDIA GeForce RTX 2080 GPU to speed up network queries for the Swing Phase Discretization approach. The neural networks are trained with Keras 2.2.4, and queried with Tensorflow 1.4 C++ API. The robot model we use is a Sarcos Humanoid robot.

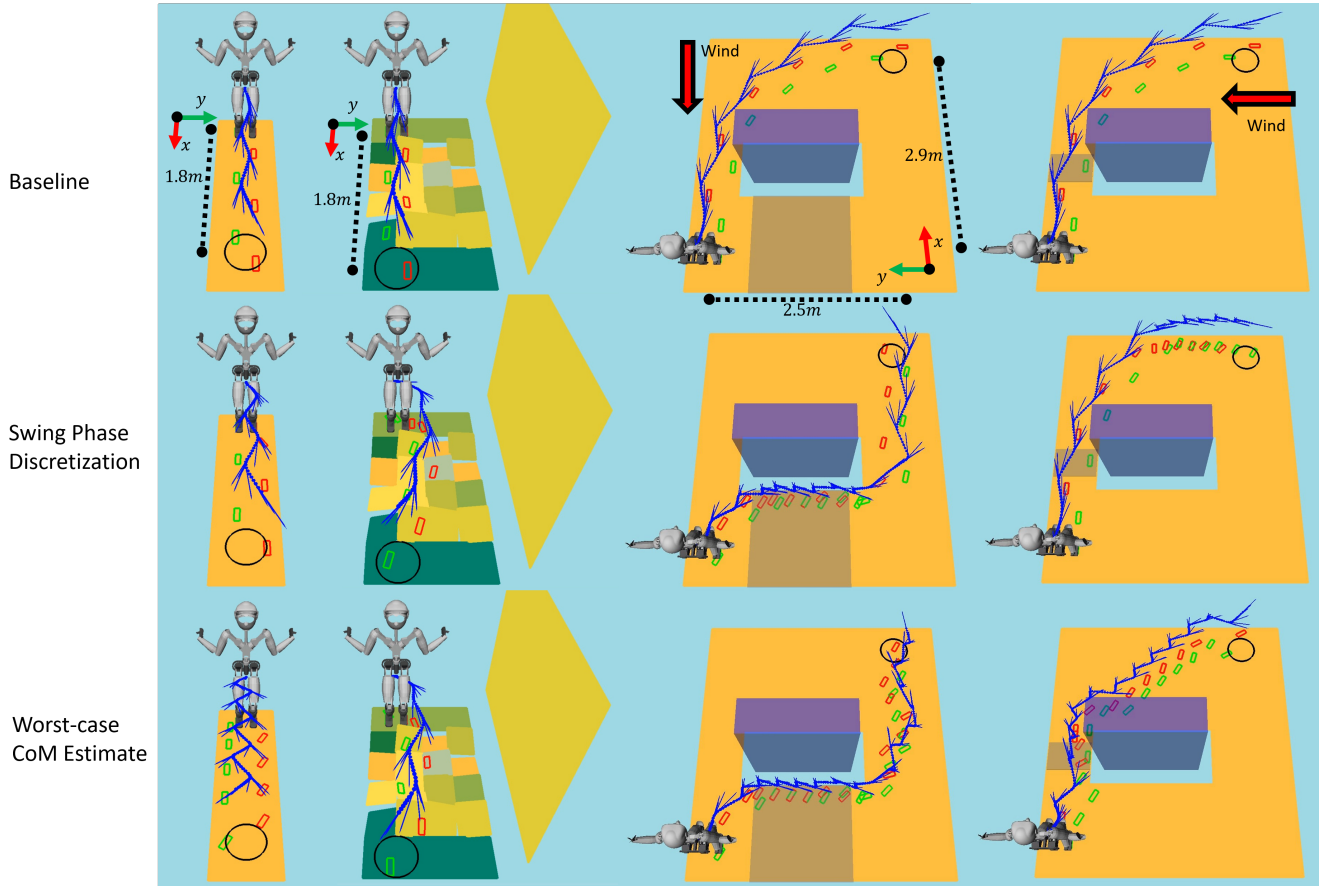


Fig. 5. From left to right: The planned footstep sequence in the narrow flat corridor, the rubble with wall, and the oil platform (wind in $-X$ and $+Y$ directions). The CoM trajectories returned by the kino-dynamic optimizer given the footstep sequences are shown in blue.

Index	Precision	Recall	Accuracy
0	97.4%	98.3%	97.8%
1	98.0%	98.0%	98.0%
2	95.9%	94.3%	95.2%
3	92.2%	90.3%	91.3%

Fig. 6. The neural networks' performance

A. Prediction of Zero-Step and One-Step Capturability

Figure 6 summarizes the performance of the neural networks in predicting capturability given an initial stance, a CoM position and a linear momentum. For each capture motion category, we train the network with 10^5 examples, and test it with another 1000 examples. Although all models perform well in predicting the capturability, the performance of predicting capture motions using palm contacts is worse than its counterpart using foot contact. This may be because capture motions using palm contacts are more likely to violate kinematic constraints and have higher variance in kinematic state, which cause them to be harder to learn.

B. Narrow Flat Ground Test Environment

In this test environment, we would like to show an intuitive result of how the robot can adjust its footstep placement to be more robust to external disturbances. We consider two lateral disturbances: $D_{\text{high}} = \{m[0, \pm 0.6, 0]^T\}$ kg \cdot m/s. In this case, the most dangerous situation is when the robot shifts its CoM to one side, and the disturbance pushes in the same direction. In this situation, the robot mainly relies on zero-step capture motion to reject the disturbance. The proposed approaches

make the robot increase the step width of the motion, which expands the support region in the y direction, and hence makes the robot more stable.

C. Rubble with Wall Test Environment

In this test, the robot has to traverse through a rubble with a side wall, similar to the rubble environment used in the DARPA Robotics Challenge. We test for five randomly generated rubble surfaces with different tilt angles, and set $D_{\text{high}} = \{m[0, 0.5, 0]^T, m[0, 0.6, 0]^T, m[0, 0.7, 0]^T, m[0, 0.8, 0]^T\}$ kg \cdot m/s. Although the wall provides a wide space for the robot to capture itself using palm contacts, it is too far away for the robot to reach if the robot simply walks straight to the goal. The planner is able to incorporate this information, and adjust the path to be close to the wall, and achieves a much more robust footstep sequence under the disturbances.

D. Oil Platform Test Environment

This test demonstrates how the planner adapts to different sets of disturbances. We consider a part of an offshore oil platform with wind blowing. There are structures on the oil platform that can block the wind, but are not suitable for palm contacts, such as electronics and pipes. We first consider $D_{\text{high}} = \{m[-0.6, 0, 0]^T, m[-0.7, 0, 0]^T, m[-0.8, 0, 0]^T\}$ kg \cdot m/s, and the wind is blocked by the structure in the center, which creates a region without disturbance, shown in grey in Figure 7. We show that the proposed

Test Environment	Approach	Number of Failed Time Step - Disturbance Pairs			Step Number	Planning Time (s) (First Solution/Best Solution within the Time Limit)	$P_{\text{success}}(T_{cp})$
		Total	Swing Phase	Double Support Phase			
Narrow Flat Ground	Baseline	36	14/40	22/60	5	0.52/0.52	83.49%
	Swing Phase Discretization	15	4/40	11/60	5	0.88/2.13	92.75%
	Worst-case CoM Estimate	18	6/72	12/108	9	0.60/4.29	91.37%
Rubble with Wall	Baseline	89.80±2.79	35.40±2.06/80	54.40±3.72/120	5±0	0.54±0.01/0.54±0.01	79.83±0.56%
	Swing Phase Discretization	17.80±7.30	16.60±5.46/128	1.20±2.39/192	8±0.63	1.17±0.84/13.74±4.58	95.84±1.65%
	Worst-case CoM Estimate	11.6±1.36	11.6±1.36/112	0±0/168	7±0	0.58±0.07/1.09±0.16	97.14±0.33%
Oil Platform (Wind in $-X$ direction)	Baseline	25	10/144	15/216	12	0.54/11.353	91.99%
	Swing Phase Discretization	0	0/288	0/432	24	1.249/22.381	100%
	Worst-case CoM Estimate	1	1/384	0/576	32	0.582/30.728	99.67%
Oil Platform (Wind in $+Y$ direction)	Baseline	45	18/144	27/216	12	0.54/11.353	86.02%
	Swing Phase Discretization	61	23/252	38/378	21	1.103/2.572	81.54%
	Worst-case CoM Estimate	42	23/312	19/468	26	0.555/35.894	86.90%

Fig. 7. The performance of each approach in all test environments. Note that there are 4 and 6 time steps in swing and double support phase, respectively. $P_{\text{success}}(T_{cp})$ is only affected by failed time step - disturbance pairs, so even some contact sequences are longer, its $P_{\text{success}}(T_{cp})$ can still be higher.

approaches leverage this region to produce low-risk contact sequences.

In another test, we consider a different wind direction with $D_{\text{high}} = \{m[0, 0.6, 0]^T, m[0, 0.7, 0]^T, m[0, 0.8, 0]^T\}$ kg · m/s. In this test, we show that the proposed approach is able to adapt to this change and produce a different contact sequence, shown in Figure 5. However, this wind direction imposes great challenges to the planner because the robot will have to travel a long distance under the strong wind. This will create many high-cost edges, which drive predicted $P_{\text{success}}(T_{cp})$ low, and many paths look similarly costly in planning. Therefore, it is not easy for ANA* to reduce search space quickly. In this case, the Worst-case CoM Estimate approach and the baseline outperform the Swing Phase Discretization approach. The first reason is that the shortest path happens to be a good path in this case. The second reason is that Swing Phase Discretization approach branches each state much slower than the other approaches due to the large amount of network queries. Therefore, it failed to find a good solution within the time limit.

E. Summary of the Planning Results

In summary, we show that the proposed approaches generate contact sequences more robust to disturbance for the scenarios considered, except for the oil platform environment with $+Y$ wind direction where Worst-case CoM Estimate approach and the baseline have similar performance. Although Worst-case CoM Estimate simplifies the capturability check of each contact transition for higher efficiency, its performance is comparable to Swing Phase Discretization approach. In general, compared to the baseline, the proposed approaches take longer to plan a contact sequence. However, if we consider scenarios with shorter horizon, such as the narrow flat ground and rubble with wall environment, Worst-case CoM Estimate approach has planning time much shorter than the execution time, and could be used in a receding horizon fashion.

X. DISCUSSION

While the proposed approaches perform much better than the baseline, there still are time steps that the robot failed to reject the disturbances when following the footstep sequence generated with the proposed approaches. In addition to wrong predictions by the network, the disturbance rejection probability model could sometimes be misleading. In Section VIII-A, we define the disturbance rejection probability to depend on the number of feasible capture poses. Since many capture poses are similar, as shown in Figure 2, if there is a wrong prediction, the network is likely to have multiple wrong predictions given by similar capture poses. To improve the model, one possible direction for future work is to use ensemble learning to increase the prediction’s robustness.

In this work, we plan humanoid contact sequences which enable the robot to more easily capture itself under external disturbances. While the decision on where to place contacts is crucial for a successful capture, CoM position and centroidal momentum also play an important role. In our current approach, during planning, the CoM position and centroidal momentum of the robot in each state is determined by a neural network which learns from a dynamics optimizer without the information of the disturbances. The solution quality may increase if we includes CoM position and centroidal momentum as decision variables in the planner. However, this will significantly increase the branching factor, and slow down the planning.

While our approach is capable of finding footstep sequences that are more robust to potential disturbances, it is still necessary to have a controller to react to disturbances during execution and select the appropriate next contact in real-time. Several approaches have been proposed to find the next contact location that helps stabilize a robot, such as in [40], but they often use a simplified model of the dynamics. It could be interesting to extend the learning part of our approach to use it in a real-time controller in order to remove the need for simplifying assumptions on the dynamics.

XI. CONCLUSION

In this paper we addressed the problem of finding contact sequences that are not only dynamically feasible but are also robust to external disturbances. It is the first time, to the best of our knowledge that a contact planning algorithm explicitly considers the effect of external disturbances. In order to enable a fast evaluation of the capturability of a transition, we trained classifiers using neural networks, leading to a significant speed-up in planning time. Experimental results demonstrate that our approach can quickly find contact plans that are less susceptible to external disturbances, which leads to more robust behaviors when executed on a real robot.

REFERENCES

- [1] S. Tonneau, A. Del Prete, J. Pettr, C. Park, D. Manocha, and N. Mansard. An efficient acyclic contact planner for multipled robots. *IEEE Transactions on Robotics*, 34(3):586–601, 2018.
- [2] K. Hauser, T. Bretl, K. Harada, and J.C. Latombe. Using motion primitives in probabilistic sample-based planning for humanoid robots. In *International Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2006.
- [3] A. Escande, A. Kheddar, S. Miossec, and S. Garsault. Planning support contact-points for acyclic motions and experiments on hrp-2. In *Experimental Robotics*, pages 293–302, 2009.
- [4] S.Y. Chung and O. Khatib. Contact-consistent elastic strips for multi-contact locomotion planning of humanoid robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.
- [5] Y.C. Lin and D. Berenson. Humanoid navigation planning in large unstructured environments using traversability-based segmentation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.
- [6] P. Fernbach, S. Tonneau, and M. Taïx. CROC: Convex resolution of centroidal dynamics trajectories to provide a feasibility criterion for the multi contact planning problem. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.
- [7] Y.C. Lin, B. Ponton, L. Righetti, and D. Berenson. Efficient humanoid contact planning using learned centroidal dynamics prediction. In *International Conference on Robotics and Automation (ICRA)*, 2019.
- [8] B. Ponton, A. Herzog, S. Schaal, and L. Righetti. A convex model of humanoid momentum dynamics for multi-contact motion generation. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2016.
- [9] A. Del Prete, S. Tonneau, and N. Mansard. Zero step capturability for legged robots in multicontact. *IEEE Transactions on Robotics*, 34(4):1021–1034, Aug 2018.
- [10] S. Wang and K. Hauser. Unified multi-contact fall mitigation planning for humanoids via contact transition tree optimization. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2018.
- [11] J. van den Berg, R. Shah, A. Huang, and K.Y. Goldberg. Anytime nonparametric A*. In *AAAI*, 2011.
- [12] J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba, and H. Inoue. Footstep planning among obstacles for biped robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2001.
- [13] J. Chestnutt, J. Kuffner, K. Nishiwaki, and S. Kagami. Planning biped navigation strategies in complex environments. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2003.
- [14] P. Michel, J. Chestnutt, J. Kuffner, and T. Kanade. Vision-guided humanoid footstep planning for dynamic environments. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2005.
- [15] L. Baudouin, N. Perrin, T. Moulard, F. Lamiriaux, O. Stasse, and E. Yoshida. Real-time replanning using 3d environment for humanoid robot. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2011.
- [16] A. Hornung, A. Dornbush, M. Likhachev, and M. Bennewitz. Anytime search-based footstep planning with suboptimality bounds. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2012.
- [17] O. Kanoun, E. Yoshida, and J. P. Laumond. An optimization formulation for footsteps planning. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2009.
- [18] R. Deits and R. Tedrake. Footstep planning on uneven terrain with mixed-integer convex optimization. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2014.
- [19] Y.C. Lin and D. Berenson. Using previous experience for humanoid navigation planning. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2016.
- [20] B. Aceituno-Cabezas, H. Dai, J. Cappelletto, J.C. Grieco, and G. Fernandez-López. A mixed-integer convex optimization framework for robust multilegged robot locomotion planning over challenging terrain. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.
- [21] B. Aceituno-Cabezas, C. Mastalli, H. Dai, M. Focchi, A. Radulescu, D. G. Caldwell, J. Cappelletto, J.C. Grieco, G. Fernandez-Lpez, and C. Semini. Simultaneous contact, gait, and motion planning for robust multilegged locomotion via mixed-integer convex optimization. *IEEE Robotics and Automation Letters*, 3(3):2531–2538, July 2018.
- [22] T. Koolen, T. de Boer, J. Rebula, A. Goswami, and J. Pratt. Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models. *The International Journal of Robotics Research*, 31(9):1094–1113, 2012.
- [23] T. Sugihara. Standing stabilizability and stepping maneuver in planar bipedalism based on the best com-zmp regulator. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2009.
- [24] T. Takenaka, T. Matsumoto, T. Yoshiike, and S. Shirokura. Real time motion generation and control for biped robot -2nd report: Running gait pattern generation-. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2009.
- [25] M. Morisawa, S. Kajita, F. Kanehiro, K. Kaneko, K. Miura, and K. Yokoi. Balance control based on capture point error compensation for biped walking on uneven terrain. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2012.
- [26] J. Engelsberger, C. Ott, and A. Albu-Schffer. Three-dimensional bipedal walking control based on divergent component of motion. *IEEE Transactions on Robotics*, 31(2):355–368, April 2015.
- [27] R.J. Griffin, G. Wiedebach, S. Bertrand, A. Leoneasa, and J. Pratt. Walking stabilization using step timing and location adjustment on the humanoid robot, atlas. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.
- [28] M. Khadiv, A. Herzog, S.A.A. Moosavian, and L. Righetti. Step timing adjustment: A step toward generating robust gaits. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2016.
- [29] J.E. Pratt and S.V. Drakunov. Derivation and application of a conserved orbital energy for the inverted pendulum bipedal walking model. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2007.
- [30] O. E. Ramos and K. Hauser. Generalizations of the capture point to nonlinear center of mass paths and uneven terrain. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2015.
- [31] T. Koolen, M. Posa, and R. Tedrake. Balance control using center of mass height variation: Limitations imposed by unilateral contact. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2016.
- [32] S. Caron, A. Escande, L. Lanari, and B. Mallein. Capturability-based pattern generation for walking with variable height. *IEEE Transactions on Robotics*, pages 1–20, 2019.
- [33] A. Herzog, S. Schaal, and L. Righetti. Structured contact force optimization for kino-dynamic motion generation. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.
- [34] D. Orin, A. Goswami, and S.H. Lee. Centroidal dynamics of a humanoid robot. *Autonomous Robots*, 35(2-3):161–176, October 2013.
- [35] B. Ponton, A. Herzog, A. Del Prete, S. Schaal, and L. Righetti. On time optimization of centroidal momentum dynamics. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.
- [36] P. Wieber. Viability and predictive control for safe locomotion. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008.
- [37] H. Herr and M. Popovic. Angular momentum in human walking. In *Journal of Experimental Biology*, pages 467–481, 2008.
- [38] J. Schulman, J. Ho, A. Lee, I. Awwal, H. Bradlow, and P. Abbeel. Finding locally optimal, collision-free trajectories with sequential convex optimization. In *Robotics: Science and Systems (RSS)*, 2013.
- [39] Y.C. Lin and D. Berenson. Humanoid navigation in uneven terrain using learned estimates of traversability. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2017.
- [40] S. Mason, N. Rotella, S. Schaal, and L. Righetti. An MPC Walking Framework With External Contact Forces. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.