Noname manuscript No.
(will be inserted by the editor)

Dynamic Balance Preservation and Prevention of Sliding for Humanoid Robots in the Presence of Multiple Spatial Contacts

Milutin Nikolić · Branislav Borovac · Mirko Raković

Received: date / Accepted: date

Abstract The main indicator of dynamic balance is the ZMP. Its original notion assumes that both feet of the robot are in contact with the flat horizontal surface (all contacts are in the same plane) and that the friction is high enough so that sliding would not occur. With increasing capabilities of humanoid robots and the higher complexity of the motion that needs to be performed, these assumptions might not hold. Having in mind that, the system is dynamically balanced if there is no rotation about the edges of the feet and if the feet do not slide, we propose a novel approach for testing the dynamic balance of bipedal robots, by using linear contact wrench conditions compiled in a single matrix (Dynamic Balance Matrix). The proposed approach has wide applicability since it can be used to check the stability of different kinds of contacts (including point, line and surface) with arbitrary perimeter shapes. Motion feasibility conditions are derived on the basis of the conditions that the wrench of each contact has to satisfy. The approach was tested by simulation in two scenarios: biped climbing up and walking sideways on the inclined flat surface which is too steep for a regular walk without additional support. The whole-body motion was synthesized and performed using a generalized task prioritization framework.

Keywords Humanoid robots · Contact stability · Whole-Body Motion

1 Introduction

The main indicator of dynamical balance, ZMP, was introduced by M. Vukobratović and his closest associates [33,35,32]. The notion of ZMP assumes that during walking the feet of the robot are in contact with the horizontal ground

This work was funded by the Ministry of Science and Technological Development of the Republic of Serbia in part under contract TR35003 and in part under contract III44008.

M. Nikolić, B. Borovac, M. Raković
Faculty of Technical Sciences, University of Novi Sad, Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia
Tel.: +381-21-4852164
E-mail: milutinn@uns.ac.rs

surface and that the friction is high enough so that sliding between the feet and the ground does not occur. Also, it has been assumed that there are no other contacts with the environment. The existence of dynamic balance has been provided by verifying that the position of the ZMP is inside the support area. If the ZMP is within, but not on the edges of the support area, the robot will not start overturning by rotating about the edge of the support area.

Several authors have tried to generalize the ZMP notion. Harada et al. [15] proposed generalized ZMP ($GZMP$) to take into account cases when there are additional contacts, that are not on the ground surface, like in the case when hands and feet of the robot are in contact with the environment simultaneously. To check if the intended motion can be performed, $GZMP$ needs to be within the region on the ground surface, defined on the basis of the mass of the robot and shape of the convex hull of all contact points. Since the contacts can be distributed spatially, the convex hull of the contacts is a 3D polyhedron. If $GZMP$ falls outside that region, the robot will lose dynamic balance and start rotating about one edge of the polyhedron. The same authors analyzed the influence of additional contact (i.e., grasping the environment) on the position of ZMP[14], and exploited that contact in order to enable the robot to climb up the high step by holding the handlebar.

In [30] the authors have used a special representation of a force screw - "wrench", where force and moment are parallel to each other. They have defined the feasible solution of wrench (FSW) and used it to check if the motion is feasible. Gravitational and inertial forces have to be counter-balanced by contact forces so that their sum has to be in the space of feasible contact forces. That condition can be used to test existence of dynamic balance when contacts between the feet and the ground are not in the same plane and when the robot's hands are in contact with the environment. Also, the authors have addressed the issue of low friction contact by introducing another criterion which can determine whether planned motion is feasible. However, it does not reveal if the sliding is going to occur at a specific contact point. Recently, Caron et al. [4] introduced a closed-form formulae for the contact wrench cone for rectangular support areas. These formulae have allowed a fast computation when creating humanoid motions in the single support phase. This method is applicable only to rectangular symmetrical robotic feet. However, it lacks the procedure to generalize to custom foot shapes. Also, the authors did not discuss the applicability of the method when the robot is in the double support phase.

1.1 Whole-body motion synthesis in presence of contact constraints

Biped walking has usually been performed on a flat ground surface. In case it becomes irregular (e.g., walking on inclined surfaces, climbing up the ladder) additional contacts of the limbs with the environment may be needed. However, each of those contacts may have specific requirements and therefore constraints must be imposed on each contact force when planning whole-body motion. Sentis and Khatib [27, 28] have created a task-prioritization framework capable of synthesizing statically balanced whole-body motion when only contact between the feet of the robot and the ground is present. In the case where multiple contacts are present,

internal forces might be needed so that the contacts remain stable[1]. In [29], the same authors addressed that issue by employing a virtual linkage model, in order to find the required internal forces which will maintain stability of all contacts. Let us emphasize that the authors introduced a requirement stating all contact forces have specific predetermined values. This unnecessarily shrinks the space of possible motion, since contact forces do not need to be kept at some specific value, but only maintained in a certain range. Recent developments [24] enabled us to impose inequality-type tasks to a prioritization framework, which was used to synthesize walk on slippery terrain [23]. This approach was validated in [16] where a slightly different version of the framework was applied on a real torque controlled robot. The authors have used a momentum control approach in order to control the behavior of the center of mass (CoM) of the robot, and later extended their approach [17] in order to synthesize feasible trajectories for the center of mass, when the contact sequence is given in advance. In [9] the authors have created a three-layer control structure. The top layer was used to plan foot-step positions. The middle layer was used to generate trajectories of the CoM and the feet. The bottom layer used the equality and inequality constrained quadratic optimization approach to calculate joint torques. The procedure was computationally expensive and in order to minimize the computational time, a commercially available library CVXGEN [22] was used to generate a code which solved the quadratic program (QP). In [19] the authors have created a QP controller for dynamic walking, and as well as a custom active set QP solver which outperformed CVXGEN 10 times in the given scenario.

Several authors have employed global optimization with inequality type constraints in order to obtain complex movements. Dai, et al. [5] have used simplified dynamics and full kinematics in order to create feasible motion. That method is unique in the sense that there is no need to schedule contact sequence in advance. Complementarity constraints are introduced in the system, so the motion, as well as contact sequence, is obtained as a result of the nonlinear optimization procedure. In [20] the authors have addressed the problem of whole-body motion synthesis by employing semi-definite programming to minimize driving torque and jerk. Tasks that have to be performed and contact constraints are introduced in the optimizing procedure as equality and inequality-type constraints. The results were verified on a real HRP-2 humanoid robot with a number of complex tasks. The main problem of the last two approaches is that they are time-consuming and thus cannot be applied on-line.

1.2 Limitations of state of the art methods

In [16, 17, 22, 19, 5, 20, 36] the contact between the foot and the ground was considered as four separate point contacts, positioned at the corners of the rectangular feet. In order to maintain stable contact, all four forces have to be within their respective friction cones. The friction cones were approximated with the four-faced

[1] In this context 'stable contact' represents the situation where there is no relative motion between the bodies in contact. In our opinion term 'stable contact' is not the most suitable to describe the state of the contact when there is no sliding and no separation. It seems that terms such as 'reliable, sustainable' are more descriptive.

pyramids, which can be overly conservative since the volume of the space of feasible forces is reduced by 36%, compared to the full friction cones. Also, that introduces additional 12 variables per surface contact into an optimization scheme which makes the whole procedure slower.

In [3] the authors have described how to calculate the contact wrench cone of the surface contact as well as the gravito-inertial wrench cone using a double descriptor method [11]. The biggest problem of this method is that the algorithm is NP-complete which limits the number of faces of the friction cones (thus making it conservative) as well as the number of points on the contact surface (thus making it inapplicable for complex foot shapes). In [3] the authors have applied this method to a robot climbing up the stairs. For a double-support phase with rectangular feet and 4-faced friction cones, it took approximately $4.5ms$ just to calculate the gravito-inertial wrench cone (GIWC). The lowest applicable sampling rate for torque controlled robots is 200Hz, so $4.5ms$ for calculating GIWC might be too high for the control of the real robot since only that single part occupies most of the time available for calculating referent control torques.

For the case when multiple contacts are present, several papers [3, 18] have derived the cone in which the gravito-inertial wrench (sum of inertial and gravitational wrenches) needs to lie in order for the motion to be feasible. Those methods are great in terms of simplicity of use since only one matrix multiplication needs to be performed in order to check if the desired motion is feasible, but on the other hand, it is time-consuming to recompute the gravito-inertial wrench cone (GIWC) at each time instance. Sometimes in order to perform the desired motion, internal forces need to be added to maintain all contacts stable, but the GIWC notion hides that information.

1.3 Contribution

In this paper we propose a novel form of contact wrench conditions written in matrix form - Dynamic Balance Matrix (DBM). It provides both necessary and sufficient condition for contact stability and is applicable to all types of contacts: point, line and surface contacts with arbitrary perimeter shapes. By carefully considering the contact mechanics it will be shown here, how the cone, in which the contact wrench of the planar contact must lie, can be calculated in polynomial time. The cone would be derived from a convex hull of a set of specially defined points in five-dimensional space. The DBM is constant when expressed in a coordinate frame of the segment which makes contact with the environment (i.e. robot's foot), so it does not need to be recalculated at time step. That makes the proposed approach computationally efficient and applicable to contact surfaces with arbitrary shapes with an arbitrary number of friction cone faces. Also, by imposing constraints on wrenches instead of forces, the number of introduced variables per surface contact drops to 6 which makes the procedure even more efficient. The influence of the shape of the contact surface, the number of sides of friction cone and orientation of the cone on the DBM will be investigated. It will be shown, as well, how the frictional forces modulate the effective shape and size of the foot.

The condition which needs to be fulfilled in order for the desired motion to be feasible is presented also. It explicitly takes into account contact stability, thus giving information about needed internal loads, without adding too much complexity.

For the case when the motion is not feasible, the procedure for determining the CoM offset so that the desired motion becomes feasible is shown. In case that the offset cannot be found the robot has to create additional contacts with the environment in order to make desired motion feasible. One method for planning contact point sequence is described in [7]. Incorporating that into a generalized task prioritization framework [24] enabled us to generate complex multi-contact dynamically balanced whole-body motions. After interfacing qpOASES [10] as a QP solver for MATLAB we were able to lower the reference control torque calculation time to $5ms$. That is a promising result and gives us assurance that this method could be employed for on-line whole-body motion synthesis. Building a custom active set QP solver might yield further decrease in calculation time as reported in [19].

The paper is organized as follows: in section 2 constraints imposed on contact forces and wrenches are analyzed and the notion of *DBM* is introduced. This is the main contribution of the paper. In chapter 3 the procedures for determining if the desired motion is feasible and if CoM can be shifted so that desired motion becomes feasible are presented. For the consistency of the paper, in section 4, the framework for whole-body motion synthesis is briefly presented. The results of the simulations where the task prioritization framework in conjunction with *DBM* is used to create motion are presented in section 5. Conclusions are given in section 6.

2 Dynamic balance matrix

Sustaining contact(s) between the robot and the environment imposes the constraints on contact wrenches. To ensure contact stability, wrenches have to be within a certain range and this has to be taken into account when planning the motion. If the motion requires wrenches that are outside possible range, this indicates the contact will not remain stable during the execution of the motion.

2.1 Point contact

Contacts between segments of the robot and the environment are unilateral. This means that segments of the robot can only push against the environment but not pull it.[2] At the point of contact, the coordinate frame will be affiliated, whose z axis will be normal to the contact surface. The condition for single point contact can be written in the following form:

$$F_z \geq 0. \tag{1}$$

Besides, we will assume that Columb friction exists between the bodies in contact. Thus, the forces in the tangential plane must satisfy:

$$F_x^2 + F_y^2 \leq \mu^2 F_z^2, \tag{2}$$

[2] One exception appears to be when the robot grasps an environment, so it can both push and pull. When looking at the grasp, it is actually comprised of several unilateral contacts, positioned in such a way that all directions of contact forces are possible. The only real exception of this assumption is when there is adhesion between the robot and the environment

where μ represents the friction coefficient. These non-linear constraints imply that the contact force must be inside the friction cone, as depicted in Fig. 1.

This constraint is quadratic and will be linearized by approximating the friction cone with a m-faced friction pyramid (Fig. 1). By increasing the number of faces of the pyramid, the approximation would be better, but that would also increase the computation complexity. After approximation, constraints (1) and (2) can be represented by a single linear condition:

$$\mathbf{S}_{\mu\ m\times 3}\mathbf{F}_{3\times 1} \succeq \mathbf{0}_{m\times 1} \tag{3}$$

where each column of matrix $\mathbf{S}_\mu$ represents one surface of the friction pyramid and has the form:

$$\mathbf{S}_{\mu k} = \left[\sin\tfrac{2k\pi}{m}\ -\cos\tfrac{2k\pi}{m}\ \mu\right];\ k = 0\ldots m-1. \tag{4}$$

This is the *face* representation of the friction pyramid and this form is well-suited for enforcing constraints on the contact forces when planning multi-contact whole-body motion. The condition (3) imposed on contact force encompasses both unilaterality of the point contacts as well as frictional constraints.

On the other hand, in order to derive the contact wrench cone for the surface contact, the friction pyramid would be rewritten in the *span* form. If we take a look at the friction pyramid it can be noted that all forces that lie within the friction pyramid can be written as a linear combination of pyramid's edges. That can be written as:

$$\mathbf{F} = \mathbf{u}_1\alpha^1 + \mathbf{u}_2\alpha^2 + \cdots + \mathbf{u}_m\alpha^m = \mathbf{U}\boldsymbol{\alpha},\ \ \alpha^i \geq 0 \tag{5}$$

where:

$$\mathbf{u}_i = \begin{bmatrix} \mu\cos\frac{2\pi(i-1/2)}{m} \\ \mu\sin\frac{2\pi(i-1/2)}{m} \\ 1 \end{bmatrix}, \mathbf{U} = \left[\mathbf{u}_1\ \mathbf{u}_2\ \ldots\ \mathbf{u}_m\right] \tag{6}$$

and $\boldsymbol{\alpha} = \left[\alpha^1\ \alpha^2\ \ldots\ \alpha^m\right]^T$

2.2 Planar contact

Most of the time, contact between the foot of the robot and the ground can be considered as planar. Constraint on the contact wrench acting on the robotic foot, which needs to be fulfilled in order to ensure stability of the contact, will be derived using span representation (5) of the friction pyramid. Sliding conditions are included and this model also considers the "yaw friction" (the torque generated by friction in the direction of a normal axis z [4]), which is crucial for avoiding undesirable yaw rotations of the robot's feet.

In general, surface contact will be called any contact between an arbitrary curved ground and planar foot where at least three non collinear point contacts exist (Fig.2). The surface contact of a single foot and the ground is modeled by multiple separate point contacts placed at each corner of the foot (for a rectangular foot, four contact points placed at the corners). It is important to note that the proposed procedure for deriving contact wrench conditions can be applied when

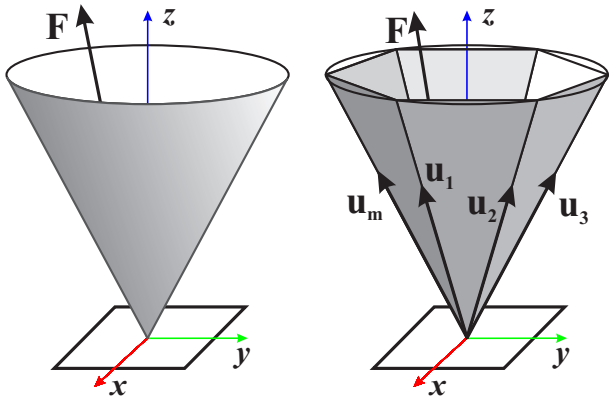


Fig. 1: The friction cone (right) and the approximating friction pyramid with generating vectors (left)

a number of contact points are greater than one. When there are two contact points we have line contact. If we have 3 or more non-collinear contact points - there is surface contact. The convex hull of the contact points defines the contact perimeter. To show that the proposed procedure does not require any shape of the foot, we will model it as a pentagon of arbitrary shape (Fig. 2).

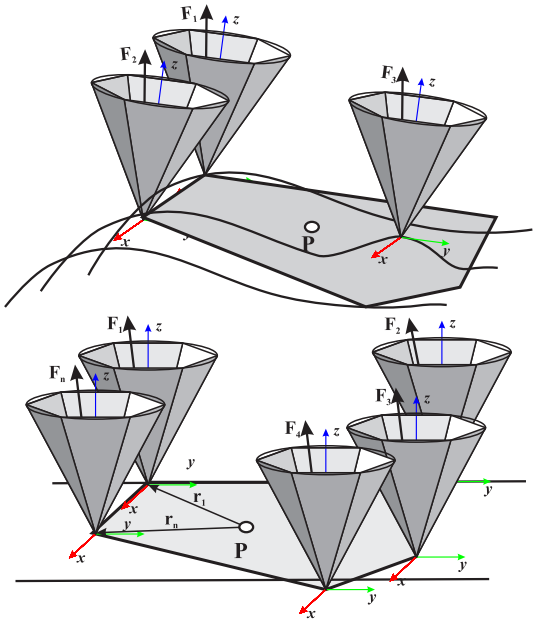


Fig. 2: Illustration of approximation of planar contact as multiple point contacts. Top figure: case when foot is only partially in contact with the environment, Bottom figure: case when the complete foot is in contact with the ground

Total contact wrench acting on the robot foot, calculated for the referent point P (inside support area) is:

$$\begin{bmatrix}\mathbf{F}_P\\ \mathbf{M}_P\end{bmatrix} = \begin{bmatrix}\sum_{i=1}^{n}\mathbf{F}_i\\ \sum_{i=1}^{n}\mathbf{r}_i\times\mathbf{F}_i\end{bmatrix} = \begin{bmatrix}\sum_{i=1}^{n}\mathbf{U}\boldsymbol{\alpha}_i\\ \sum_{i=1}^{n}[\mathbf{r}_i]_\times\mathbf{U}\boldsymbol{\alpha}_i\end{bmatrix}. \tag{7}$$

Vectors $\mathbf{r}_1$ to $\mathbf{r}_n$ represent positions of contact points relative to referent point P. $[\mathbf{u}]_\times$ is a skew-symmetric cross product operator matrix. Matrix $\mathbf{U}$ can be also written in the form:

$$\mathbf{U} = \begin{bmatrix}\mu\cos\frac{\pi}{m} & \dots & \mu\cos\frac{\pi(2m-1)}{m}\\ \mu\sin\frac{\pi}{m} & \dots & \mu\sin\frac{\pi(2m-1)}{m}\\ 1 & \dots & 1\end{bmatrix} = \begin{bmatrix}\mu\mathbf{C}\\ \mu\mathbf{S}\\ \mathbf{1}\end{bmatrix}, \tag{8}$$

where $\mathbf{1}$ represents row vector of length m with all elements 1. After introducing (8) in (7) and calculating the cross product it can be obtained:

$$\begin{bmatrix}\mathbf{F}_P\\ \mathbf{M}_P\end{bmatrix} = \begin{bmatrix}\mu\mathbf{C}\sum_{i=1}^{n}\boldsymbol{\alpha}_i\\ \mu\mathbf{S}\sum_{i=1}^{n}\boldsymbol{\alpha}_i\\ \mathbf{1}\sum_{i=1}^{n}\boldsymbol{\alpha}_i\\ \sum_{i=1}^{n}y_i\mathbf{1}\boldsymbol{\alpha}_i\\ \sum_{i=1}^{n}-x_i\mathbf{1}\boldsymbol{\alpha}_i\\ \sum_{i=1}^{n}(-y_i\mu\mathbf{C}+x_i\mu\mathbf{S})\boldsymbol{\alpha}_i\end{bmatrix} \tag{9}$$

It can be noted that the normal force is the sum of all alphas $F_z = \sum_{i=1}^{n}\sum_{j=1}^{m}\alpha_i^j$. Since all alphas are greater or equal to zero ($\alpha_i^j \geq 0$) it can be easily deduced that normal force has to be greater or equal to zero $F_z \geq 0$. We will remove the vertical force from the wrench vector, so all five other components can be written in the following form:

$$\begin{bmatrix}F_x\\ F_y\\ M_x\\ M_y\\ M_z\end{bmatrix} = \mathbf{P}\begin{bmatrix}\boldsymbol{\alpha}_1\\ \boldsymbol{\alpha}_2\\ \vdots\\ \boldsymbol{\alpha}_n\end{bmatrix}_{(nm)\times 1} \tag{10}$$

$$\mathbf{P} = \begin{bmatrix}\mu\mathbf{C} & \dots & \mu\mathbf{C}\\ \mu\mathbf{S} & \dots & \mu\mathbf{S}\\ y_1\mathbf{1} & \dots & y_n\mathbf{1}\\ -x_1\mathbf{1} & \dots & -x_n\mathbf{1}\\ -y_1\mu\mathbf{C}+x^1\mu\mathbf{S} & \dots & -y_n\mu\mathbf{C}+x_n\mu\mathbf{S}\end{bmatrix}_{5\times(nm)} \tag{11}$$

When both sides of the equation are divided by F_z we obtain:

$$\mathbf{p}_{-z} = \begin{bmatrix}F_x/F_z\\ F_y/F_z\\ M_x/F_z\\ M_y/F_z\\ M_z/F_z\end{bmatrix} = \mathbf{p}_1\frac{\alpha_1^1}{F_z} + \mathbf{p}_2\frac{\alpha_1^2}{F_z} + \dots + \mathbf{p}_{n\times m}\frac{\alpha_n^m}{F_z}, \tag{12}$$

where $\mathbf{p}_{-z}$ represents 5D vector obtained from the contact wrench by removing normal force F_z and dividing vector by F_z, $\mathbf{p}_i$ represents point in 5D space which

corresponds to i-th column of matrix $\mathbf{P}$. Since $F_z \geq 0$ and $\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{\alpha_i^j}{F_z} = 1$, $\alpha_i^j \geq 0$ it can be concluded that the left hand side is a convex combination of the points $\mathbf{p}_i$, thus the vector $\mathbf{p}_{-z}$ *must lie within the convex hull of the points* $\mathbf{p}_i$.

The procedure for calculating the convex hull of points in hyperspace is well known and has low complexity. For the Quickhull algorithm implemented in Matlab the complexity is $O\left(n_{points}\log n_{points}\right)$[2], which reduces to $O\left(nm\log\max(n,m)\right)$ for the case considered here. As a result of the Quickhull the hyperplanes that bound the convex hull are obtained. Each of the hyperplanes introduces one inequality constraint that can be written in the form:

$$v_1\frac{F_x}{F_z} + v_2\frac{F_y}{F_z} + v_3\frac{M_x}{F_z} + v_4\frac{M_y}{F_z} + v_5\frac{M_z}{F_z} \geq w \tag{13}$$

where $v_1 \ldots v_5$ and w are parameters of the hyperplane. After multiplying both sides with F_z and rearranging the terms in the equation, it can be seen that one hyperplane from the convex hull introduces one inequality constraint on the wrench acting on the referent point P:

$$\begin{bmatrix} v_1 & v_2 & -w & v_3 & v_4 & v_5 \end{bmatrix}\begin{bmatrix}\mathbf{F}_P \\ \mathbf{M}_P\end{bmatrix} \geq 0 \tag{14}$$

Each hyperplane introduces one constraint in the previous form, so after stacking all those constraints together with $F_Z \geq 0$ in one matrix it can be written:

$$\mathbf{Z}\begin{bmatrix}\mathbf{F}_P \\ \mathbf{M}_P\end{bmatrix} \succeq 0 \tag{15}$$

where $\mathbf{Z}$ represents dynamic balance matrix - *DBM*. It will be proven later that this is a necessary and sufficient condition for surface contact stability, which includes unilaterality of the contact as well as sliding. It needs to be emphasized here that DBM is constant with respect to the local coordinate frame of the robot's feet, so it needs to be computed only once for a given foot shape. If only the part of the foot is in contact with the ground the corresponding DBM has to be recomputed taking into account the new contact perimeter.[3]

2.3 Surface contact stability

It will be shown here that condition (15) represents a necessary and sufficient condition for the *weak stability* of the surface contacts. We will consider only *weak stability* since it is impossible to determine the strong stability in general if the sufficient friction assumption is removed as stated in [18]. The definition of *weak stability* under Columb friction [26] widely used in robotics literature is:

Definition 1. *For a given external force* $\mathbf{F}$ *contact is weakly stable if a valid contact force exists that induces zero body accelerations.*

[3] An on-line demo showing calculation of the DBM can be found at the author's website `www.milutinnikolic.info/dbm/`

Contact force is said to be valid if it fulfills the conditions (1) and (2). Surface contacts are a continuum of infinitesimal point contacts, where contact forces at each infinitesimal point are defined by a pressure field (for normal forces) and a stress field (for tangential forces). If the pressure and stress are Dirac fields, it is sufficient to consider only point contacts at the corners of the convex hull of the contact patch instead of whole surface [4]. Proposition 1 from [3] states that contact wrench $\mathbf{w}$ is valid if and only if there exists valid contact force at the vertices of the polygon that sum up to $\mathbf{w}$. If the contact wrench is valid the contact would be weakly stable.

Theorem 1. *The given external wrench* $\left[\mathbf{F}^T\ \mathbf{M}^T\right]^T$ *is valid if and only if the inequality* (15) *is fulfilled*

Proof. Firstly, it needs to be proven if the contact forces at the vertices of the convex hull are valid then the contact wrench fulfills equation (15). Proof of this part comes from the procedure for deriving (15).

Conversely, assume that $\left[\mathbf{F}^T\ \mathbf{M}^T\right]^T$ fulfills condition (15). In that case $F_z \geq 0$ and point $\mathbf{p}_{-z}$ lies within the convex hull of points $\mathbf{p}_i$. As a result point $\mathbf{p}_{-z}$ can be represented as a convex combination of points $\mathbf{p}_i$, meaning that $\mathbf{p}_{-z} = \sum_{k=1}^{n\times m} \beta_k \mathbf{p_k}$ such that $\beta_k \geq 0$. By multiplying β_k-s with vertical force F_z we can obtain α_i^j and then reconstruct valid forces that sum up to $\left[\mathbf{F}^T\ \mathbf{M}^T\right]^T$ using (5). □

2.4 Properties of DBM

In this section we will briefly discuss the influence of the shape of the foot, number of faces of the friction cone and friction cone orientation on the number of faces of the convex hull and thus the number of rows in the DBM. First, we will consider how the number of faces influences the volume of allowable forces. When the friction cone is approximated with a regular 4-faced pyramid, the pyramid has only 64% of the cone volume, making the volume of the space of allowed forces after approximation just 64% of the volume of the whole space. Also in some directions tangential force is too restricted and in the worst case the tangential force can not exceed 70.7% of the maximal tangential force. That might be too restrictive and in order to create less conservative conditions, better approximation of the friction cone is needed, which is achieved by increasing the number of faces of the pyramid. In Table 1 the relationship between the number of faces of the friction pyramid and the percentage of the volume included along with a percentage of the maximal force is shown. It can be seen that in order to have at least 90% of the volume covered and at least 90% of max force in all directions, approximation by a minimum 8 faces is required.

The increase in the number of faces of the friction cone strongly influences the number faces of convex hull, as seen in Table 1. Also, the shape of the foot and type of the friction cone influence the number of faces in convex hull. Two types of the foot will be considered here, shown in Fig. 3, a standard rectangular foot and irregular 5-sided foot which resembles the shape of human feet more closely. Also, two types of friction cone orientations will be considered. Type A friction

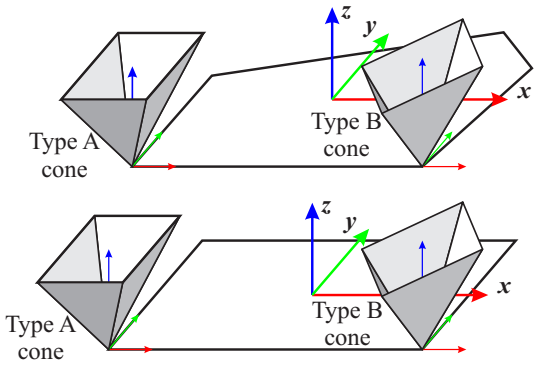


Fig. 3: 5-sided and rectangular foot with Type A and Type B friction cones

cone is where faces of the cone are parallel with the x or y axes, thus parallel to the edges of the rectangular feet; Type B friction cone is rotated so that the edges of the cone lie in the same vertical plane as the x or y axes. Friction cone defined by eqns. (5) and (6) is the friction cone of Type A. It can be seen that type of cone only influences the total number of faces for the rectangular foot when the number of faces is divisible by 4. If the foot is irregular, the symmetry is broken and the number of faces of the convex hull grows substantially. On the other hand the lowest number of faces is obtained for a rectangular foot and Type A cone with a number of faces is divisible by 4, since in that case all edges of the foot have parallel faces in the friction pyramid.

Table 1: Volume, Max. Force and Numbers of faces

Cone Faces	% volume	% max force	Number of faces			
			Rectangular		5-sided	
			A	B	A	B
4	63.7	70.7	16	32	44	68
6	82.7	86.6	54	50	142	144
8	90	92.4	80	132	238	296
10	93.6	95.1	178	186	456	428
12	95.5	96.6	240	328	602	672

In order to illustrate the influence of the number of faces of the friction cone to the space of applicable forces, we have studied it's influence on "yaw friction", i.e., maximal torque applicable in direction of the normal z. For each position of the ZMP we have calculated the maximal applicable yaw torque and created corresponding heatmaps shown in Fig. 4. The maximal normal torque goes from -0.05 (represented by the deep blue color), passes through zero (green color) and goes up to 0.15 (represented with the dark red color). For the case depicted a unit vertical force ($F_z = 1$) and friction coefficient $\mu = 0.9$ are used. The left column corresponds to the case where the tangential force (Columb friction) is 0. It is evident (going from top to bottom) that the area of high yaw friction grows when the number of faces increases from 4 to 8 and finally to 12. In cases where the highest normal torques are required the ZMP is restricted to the center of

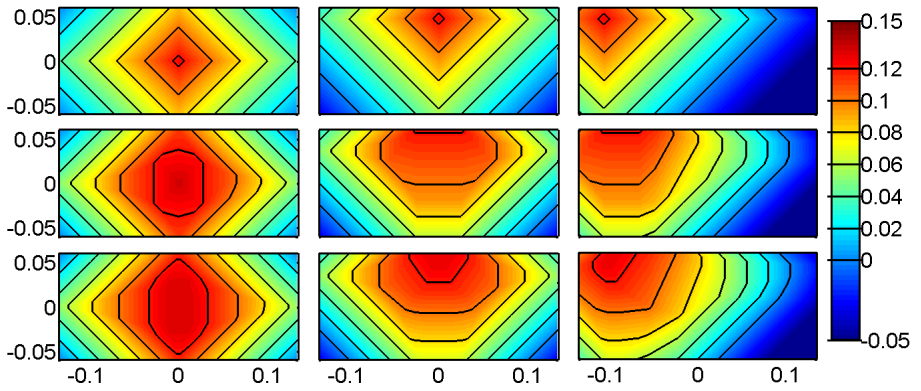


Fig. 4: On the heatmaps a rectangular foot is shown. For each position of the ZMP on the foot, maximal applicable normal torque is shown in color, starting from -0.05 (deep blue), passing through zero (green) and up to 0.15 (dark red). Normal force is 1 and friction coefficient $\mu = 0.9$. Left column: Case when the frictional forces are zero, Center column: Case when frictional force of intensity 0.5 acts in the x direction, Right column: Case when frictional force of 0.5 acts in both x and y directions. Top row: 4 faced friction cone, Middle row: 8 faced friction cone, Bottom row: 12 faced friction cone. Black lines represent the contours with normal torques from 0 to 0.12 with step 0.02

the foot when 4 faces approximation is used, but is allowed to be within some region if the less conservative condition is employed. When the friction acts in the x direction with intensity of $F_x = 0.5$ the middle column of Fig. 4 is obtained. In such an instance area of high applicable normal torque moves in the orthogonal direction towards the edge of the foot. Also, patches where maximal normal torque is negative appear at the bottom corners of the foot. That means in order for the ZMP to be in that area negative normal torque has to be applied.

That effect is even more clear when the friction force is acting in both x and y directions $F_x = 0.5$ and $F_y = 0.5$ (right column in Fig. 4). The whole bottom-right part of the foot has maximal vertical torque negative, meaning that it is impossible to have ZMP in that area without appropriate normal torque applied. That area is much bigger when a 4-faced friction cone is used, compared to the case when 8-faced and 12-faced friction cones are used. Also, it can be seen how the required frictional forces and vertical torque modulate the effective size and shape of the foot, by making it smaller when large frictional forces and 'yaw friction' is required.

2.5 System with multiple contacts

When multiple spatial contacts between the robotic system and the environment exist it is easy to generalize condition (15). An example of such a system is given in Fig. 5, where two planar contacts exist between the ground and the feet, as well as point contact between the robot's hand and vertical wall. Contact wrenches between the ground and the feet must satisfy condition (15), while contact between

the hand and vertical wall has to satisfy (3). It is important to recall that these constraints are given in local coordinate frames of the contacts. Thus, if we express contact forces in a world coordinate frame, we must pre-multiply them with corresponding rotation matrices in order to express them in a local coordinate frame of the contact. That can be written:

$$\begin{bmatrix} \mathbf{Z}_L & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Z}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{S}_\mu \end{bmatrix} \begin{bmatrix} \begin{bmatrix} \mathbf{R}_L^T \mathbf{F}_L \\ \mathbf{R}_L^T \mathbf{M}_L \end{bmatrix} \\ \begin{bmatrix} \mathbf{R}_R^T \mathbf{F}_R \\ \mathbf{R}_R^T \mathbf{M}_R \end{bmatrix} \\ \mathbf{R}_H^T \mathbf{F}_H \end{bmatrix} = \bar{\mathbf{Z}} \mathbf{F}_{ext} \succeq \mathbf{0}. \tag{16}$$

For such a case *Composite Dynamic Balance Matrix (CDBM)* $\bar{\mathbf{Z}}$ is introduced. It is a diagonal block matrix of dynamic balance matrices for left ($\mathbf{Z}_L$) and right foot ($\mathbf{Z}_R$), and friction cone ($\mathbf{S}_\mu$) for robot's hand multiplied by a diagonal block matrix consisting of rotation matrices ($\mathbf{R}_L$, $\mathbf{R}_R$ and $\mathbf{R}_H$) for each contact. The vector of generalized contact forces acting on the robot expressed in the world coordinate frame is denoted by $\mathbf{F}_{ext} = \begin{bmatrix} \mathbf{F}_L^T & \mathbf{M}_L^T & \mathbf{F}_R^T & \mathbf{M}_R^T & \mathbf{F}_H^T \end{bmatrix}^T$.

All results from this chapter are briefly summarized in table 2. For each case considered in this section the table shows:the contact type; assumptions and required knowledge about contact; resulting constraint and mechanical quantity to which the constraint is applied. It can be noticed that for all types of contacts, the orientation of the contact surface is considered to be known. Recently developed algorithm Kintinuous [37] is able to reconstruct the 3D space around the robot with resolution under $5mm$ [8] in real time, using the Microsoft Kinect commodity RGB-D sensor. Having that in mind, the position and orientation of contact surfaces can be assumed to be known. If accuracy provided by Kintinuous algorithm is not sufficient, it could be further refined by force sensing together with internal joint angle sensors and kinematic model of the robot.

Table 2: Summary of the constraints

Type of contact	Assumptions and knowns	Constraint	Applied to
Point Contact	Friction coefficient Normal to the contact surface	Friction cone expressed in coordinate frame of contact surface. Eq. (3)	Contact Force
Surface Contact	Friction coefficient Shape of contact surface All points must coplanar	DBM expressed in coordinate frame of segment in contact. Eq. (15)	Contact Wrench
Multiple Contacts	Types of all contacts Orientations of all contacts	CDBM expressed in world coordinate frame. Eq. (16)	Vector of all contact forces and torques

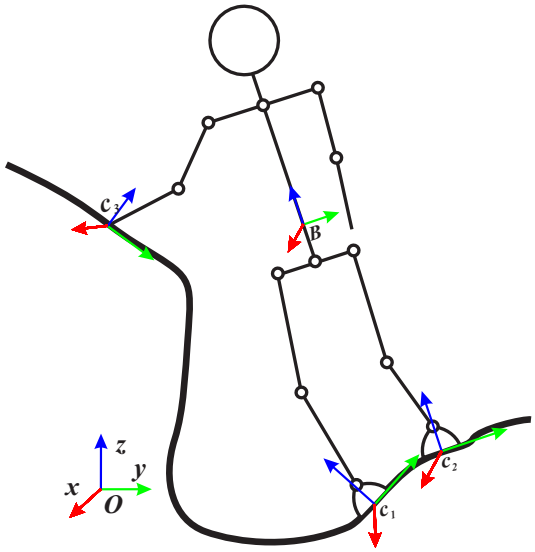


Fig. 5: System with multiple contacts. Contacts between the feet and the ground are considered planar while contact between the hand and the wall is considered as point contact

3 Motion feasibility

In complex environments, such as the one shown in Fig. 5 it is extremely important to check if the intended motion is feasible, before a robot initiates its execution. For example, a robot cannot lift its hand from the surface because the robot will lose dynamical balance and ultimately fall. Also, if the surface in contact with the hand has a low friction coefficient, the robot will be unable to maintain even static posture, since the hand will slide, causing the robot to fall.

The total wrench that all contact wrenches create for the CoM of the system from Fig. 5 is:

$$\begin{bmatrix} \mathbf{F}_C \\ \mathbf{M}_C \end{bmatrix} = \begin{bmatrix} \mathbf{F}_L + \mathbf{F}_R + \mathbf{F}_H \\ \mathbf{r}_L^{CoM} \times \mathbf{F}_L + \mathbf{M}_L + \mathbf{r}_R^{CoM} \times \mathbf{F}_R + \\ + \mathbf{M}_R + \mathbf{r}_H^{CoM} \times \mathbf{F}_H \end{bmatrix} = \mathbf{G}_C \mathbf{F}_{ext}. \tag{17}$$

The matrix $\mathbf{G}_C$ represents contact matrix calculated for the CoM. It relates all contact wrenches acting on the body with total wrench acting on the CoM. If a robot with mass m needs to perform a motion with a desired acceleration of CoM $\mathbf{a}_C^{des}$ and desired rate of change of angular momentum $\dot{\mathbf{L}}_C^{des}$, contact forces need to counterbalance gravitational and inertial forces:

$$\begin{bmatrix} m\mathbf{a}_C^{des} \\ \dot{\mathbf{L}}_C^{des} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_C \\ \mathbf{M}_C \end{bmatrix} + \begin{bmatrix} m\mathbf{g} \\ \mathbf{0} \end{bmatrix}. \tag{18}$$

Since contact wrenches are bounded, total wrench for the CoM can not assume arbitrary value, meaning that a robot is not able to perform an arbitrary desired

motion. By combining eqns (16), (17) and (18) it is easy to derive a linear program whose feasibility tells us if the desired motion is possible:

$$\begin{aligned} \text{find} \quad & \mathbf{F}_{ext} \\ \text{such that:} \quad & \mathbf{G}_C\mathbf{F}_{ext} = \begin{bmatrix} m\left(\mathbf{a}_C^{des} - \mathbf{g}\right) \\ \dot{\mathbf{L}}_C^{des} \end{bmatrix} \\ & \bar{\mathbf{Z}}\mathbf{F}_{ext} \succ \mathbf{0}. \end{aligned} \tag{19}$$

If there is a solution to the previous program, the desired motion is feasible, otherwise desired motion is not feasible. The intuition behind this program is that it tries to find the external force that produces required motion of CoM (equality constraint) so that all contacts are stable (inequality constraint). If such force cannot be found, it either means that contact will become unstable when the robot tries to perform the motion and consequently fall (inequality constraint is not fulfilled) or while maintaining all the contacts stable the robot will not be able to perform desired motion (equality constraint is not fulfilled). The solution to the problem might not be unique, but finding single external force that fulfills both constraints is enough to prove that the desired motion is feasible. Although the idea looks simple, this program is a very powerful tool when determining if some intended motion can be performed or not. In the case where desired motion is feasible, it can be easily checked if some contact can be broken so that the desired motion remains feasible by removing corresponding parts from matrices $\bar{\mathbf{Z}}, \mathbf{G}_C$ and $\mathbf{F}_{ext}$ and running the linear program (19) once more.

When desired motion is not feasible, it is possible to check if CoM can be moved from point C, whose position is given by $\mathbf{r}_C$, to some new point D whose position is given by $\mathbf{r}_D = \mathbf{r}_C + \boldsymbol{\Delta}$, so that the motion becomes feasible. Now, total wrench of contact forces for point D is:

$$\begin{bmatrix} \mathbf{F}_D \\ \mathbf{M}_D \end{bmatrix} = \begin{bmatrix} \mathbf{F}_C \\ \mathbf{M}_C \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ [\mathbf{F}_C]_\times \end{bmatrix} \boldsymbol{\Delta} \tag{20}$$

By taking into account that the force acting on CoM must be equal to the sum of gravitational and inertial forces it is easy to derive the linear program whose solution is the $\boldsymbol{\Delta}$:

$$\begin{aligned} \text{find} \quad & \boldsymbol{\Delta},\ \mathbf{F}_{ext} \\ \text{such that:} \quad & \bar{\mathbf{Z}}\mathbf{F}_{ext} \succ \mathbf{0} \\ & \mathbf{G}_C\mathbf{F}_{ext} = \begin{bmatrix} m\left(\mathbf{a}_C^{des} - \mathbf{g}\right) \\ \dot{\mathbf{L}}_C^{des} \end{bmatrix} - \begin{bmatrix} \mathbf{0} \\ \left[m\left(\mathbf{a}_C^{des} - \mathbf{g}\right)\right]_\times \end{bmatrix} \boldsymbol{\Delta} \end{aligned} \tag{21}$$

If a solution to this linear problem exists, it is possible to modify the motion so that the desired motion becomes feasible. On the other hand, if the solution to this problem does not exist a robot has to make additional contacts with the environment in order to be able to perform the intended motion. The intuition behind this program is similar as in the previous case. We are trying to find $\boldsymbol{\Delta}$ and $\mathbf{F}_{ext}$ so that external force would produce required wrench and thus required motion if CoM is at point D while maintaining all contacts stable. Selection of the right contact point is out of the scope of this paper, but once the new contact point

candidate is found it can be tested if additional contact will make movement feasible (additional contacts change vector $\mathbf{F}_{ext}$ and matrices $\mathbf{G}_C$ and $\bar{\mathbf{Z}}$) by running the program (19).

4 Whole-body motion synthesis

Up to now, the paper has been concerned with motion constraints imposed by the unilateral contacts with Columb friction. Now, the focus would be switched to way of using such constraints for synthesizing complex whole body motions. To do so, the a generalized task-prioritization framework will be employed. It is based on [27,28] and [29], and full description can be found in [24,23,25]. Although, the framework is developed as a part of previous research, the short description will be given in order to make the paper more clear and self-contained. The way of including motion constraints summarized in table 2 into task prioritization framework would be briefly described in this section.

Contacts constrain the system, which can be written as $\bar{\mathbf{J}}\dot{\mathbf{q}} = \mathbf{0}$, where the vectors of robot's joint coordinates, velocities and accelerations are $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$. $\bar{\mathbf{J}}$ represents a composite Jacobian matrix for all contacts between the robot and the environment. For the case shown in Fig. 5 it has the form $\bar{\mathbf{J}} = \left[\mathbf{J}_L^T\ \mathbf{J}_R^T\ \mathbf{J}_{Hlin}^T\right]^T$, where $\mathbf{J}_L$ and $\mathbf{J}_R$ are Jacobian matrices for the left and right foot, while $\mathbf{J}_{Hlin}$ is just a linear part of the Jacobian matrix associated with the robot's hand. Dynamics of the multi body system with contacts is given by:

$$\mathbf{H}\ddot{\mathbf{q}} + \mathbf{h_0} = \boldsymbol{\tau} + \bar{\mathbf{J}}^T\mathbf{F}_{ext}, \tag{22}$$

where $\mathbf{H}$ is the symmetric positive definite inertia matrix, $\mathbf{h}_0$ is the vector which includes the velocity and gravitational loads, and $\boldsymbol{\tau}$ is the vector of joint torques. The constraint equation $\bar{\mathbf{J}}\dot{\mathbf{q}} = \mathbf{0}$ can be differentiated with respect to time, obtaining $\bar{\mathbf{J}}\ddot{\mathbf{q}} + \dot{\bar{\mathbf{J}}}\dot{\mathbf{q}} = \mathbf{0}$. Now it is easy to combine the constraint with the dynamics of the system (22), so in case $\bar{\mathbf{J}}$ has a full row rank, external force can be calculated from:

$$\mathbf{F}_{ext} = \left(\bar{\mathbf{J}}\mathbf{H}^{-1}\bar{\mathbf{J}}^T\right)^{-1}\left(-\dot{\bar{\mathbf{J}}}\dot{\mathbf{q}} - \bar{\mathbf{J}}\mathbf{H}^{-1}\left(\boldsymbol{\tau} - \mathbf{h}_0\right)\right). \tag{23}$$

This external force must comply with (16) so that all contacts remain stable. This needs to be taken into account when synthesizing whole-body motion. For such a purpose the task prioritization framework will be employed.

The goal of the framework is to control the system given by (22) and (23) such that p tasks written in the form of:

$$\mathbf{A}_i\ddot{\mathbf{q}} = \mathbf{b}_i;\ \mathbf{A}_i\ddot{\mathbf{q}} \preceq \mathbf{b}_i;\ \text{or}\ \mathbf{A}_i\mathbf{H}^{-1}\boldsymbol{\tau} \preceq \mathbf{b}_i \tag{24}$$

are fulfilled in prioritized manner. That means that any lower priority task is executed in such a way that it does not interfere with the execution of any task with higher priority. Tasks of higher priorities are fulfilled without regard to tasks of lower priorities. Tasks written in the forms of $\mathbf{f}(t, \mathbf{q}) = \mathbf{0}$ or $\mathbf{f}(t, \mathbf{q}, \dot{\mathbf{q}}) = \mathbf{0}$ can be written in the required form (24) by differentiating them with respect to time twice or once, respectively. Similarly $\mathbf{f}(t, \mathbf{q}) \preceq \mathbf{0}$ or $\mathbf{f}(t, \mathbf{q}, \dot{\mathbf{q}}) \preceq \mathbf{0}$ can be written in the required form by expanding them up to the second derivative using the Taylor series.

Algorithm 1 Procedure for calculating control torques

$\mathbf{N}_c \leftarrow \mathbf{I}$, $\mathbf{T}_c \leftarrow \mathbf{0}$ $\mathbf{C}_a \leftarrow []$, $\mathbf{C}_\tau \leftarrow []$ $\mathbf{d}_a \leftarrow []$, $\mathbf{d}_\tau \leftarrow []$
if system with contacts **then**
  $\mathbf{N}_c := \mathbf{I} - \mathbf{B}_0^+ \mathbf{B}$, $\mathbf{T}_c := \mathbf{B}_0^+ (\mathbf{b}_0 + \mathbf{B}_0 \mathbf{p})$
end if
for every task i **do**
  if inequality **then**
    if dependent on acceleration **then**
      $\mathbf{C}_a := \begin{bmatrix} \mathbf{C}_a \\ \mathbf{B}_i \end{bmatrix}$; $\mathbf{d}_a \leftarrow \begin{bmatrix} \mathbf{d}_a \\ \mathbf{b}_i \end{bmatrix}$
    else if dependent on torque **then**
      $\mathbf{C}_\tau := \begin{bmatrix} \mathbf{C}_\tau \\ \mathbf{B}_i \end{bmatrix}$; $\mathbf{d}_\tau \leftarrow \begin{bmatrix} \mathbf{d}_\tau \\ \mathbf{b}_i \end{bmatrix}$
    end if
    find $\mathbf{u}$
    s. t.: $\ddot{\mathbf{r}} + \mathbf{p} = \mathbf{T}_c + \mathbf{N}_c \mathbf{T}_{prev} + \mathbf{N}_c \mathbf{N}_{prev} \mathbf{u}$
      $\mathbf{C}_a \ddot{\mathbf{r}} \preceq \mathbf{d}_a$
      $\mathbf{C}_\tau (\mathbf{T}_{prev} + \mathbf{N}_{prev} \mathbf{u}) \preceq \mathbf{d}_\tau$
    if system feasible **then**
      continue
    end if
  else
    minimize $\|\mathbf{B}_i \ddot{\mathbf{r}} - \mathbf{b}_i\|_2^2$
    s. t. : $\ddot{\mathbf{r}} + \mathbf{p} = \mathbf{T}_c + \mathbf{N}_c \mathbf{T}_{prev} + \mathbf{N}_c \mathbf{N}_{prev} \mathbf{u}$
      $\mathbf{C}_a \ddot{\mathbf{r}} \preceq \mathbf{d}_a$
      $\mathbf{C}_\tau (\mathbf{T}_{prev} + \mathbf{N}_{prev} \mathbf{u}) \preceq \mathbf{d}_\tau$
    $\mathbf{N}_{prev} := \mathbf{N}_{prev} \left(\mathbf{I} - (\mathbf{B}_i \mathbf{N}_c \mathbf{N}_{prev})^+ \mathbf{B}_i \mathbf{N}_c \mathbf{N}_{prev} \right)$
  end if
  $\mathbf{T}_{prev} := \mathbf{T}_{prev} + \mathbf{N}_{prev} \mathbf{u}$;
end for
$\boldsymbol{\tau} \leftarrow \mathbf{H}^{1/2} \mathbf{T}_{prev}$

Based on [24] the procedure for calculating control torques is derived and shown in Algorithm 1. For each task the optimization problem is solved and vector of control torques is updated. Task matrices are modified by post-multiplying them by the inverse of the root of inertia matrix $\mathbf{B}_i = \mathbf{A}_i \mathbf{H}^{-1/2}$. Constraint matrix is $\mathbf{B}_0 = \bar{\mathbf{J}} \mathbf{H}^{-1/2}$ and vector $\mathbf{b}_0 = -\dot{\bar{\mathbf{J}}} \dot{\mathbf{q}}$. The importance of these matrices is emphasized in [31]. The modified vector of joint acceleration is $\mathbf{r} = \mathbf{H}^{1/2} \ddot{\mathbf{q}}$ while the modified vector of velocity and gravitational effects is $\mathbf{p} = \mathbf{H}^{-1/2} \mathbf{h}_0$. Since the inertia matrix $\mathbf{H}$ is symmetric positive-definite it's root $\mathbf{H}^{1/2}$ always exists. Null space of constraint is $\mathbf{N}_C$ while modified torque induced by constraint is denoted by $\mathbf{T}_C$. Null space of all tasks preceding the current task is denoted by $\mathbf{N}_{prev}$ and the modified torque which needs to be applied in order to fulfill all tasks preceding the current one is $\mathbf{T}_{prev}$. Matrices $\mathbf{C}_a$ and $\mathbf{C}_\tau$ and vectors $\mathbf{d}_a$ and $\mathbf{d}_\tau$ are used in order to include inequality tasks in the framework.

5 Simulation results

In this section simulation results are presented where the simulation scenarios are selected to illustrate the capabilities of the proposed approach. The *CDBM* and procedures for determining feasibility of the motion and motion modification described in 3 were used in following the examples. The robot is simulated as a

floating-base articulated mechanism with total 27 joints (two 7 degree of freedom (DoF) legs, two 6DoF arms and 1 DoF head). Together with 6 DoF that describe the position of floating base, the system has total 33 DoF. The geometry of the robot as well as masses and moments of inertia of all the segments are chosen to resemble the adult male person. The simulation of floating-base mechanism is performed based on [34], while the model of rigid body with viscoelastic layer was used to simulate contacts with Columb frictions[12,13].

5.1 Climbing up the steep ramp

In this scenario a robot should climb up a steep ramp angled at 45°. Due to high inclination and finite frictional forces it is not possible for the robot to simply walk up the ramp. In order to climb over it the robot has to grasp the handlebars which are at the top of the ramp. This scenario is chosen to illustrate point, line and surface contacts, with cases of single and multiple contacts and with diverse spatial distribution of the contacts.

From a task-prioritization perspective, during the motion the robot had to perform several tasks. The task of the highest priority was to maintain joint torques between the predefined saturation limits. This can be easily written in the required form (third case in (24)). The task of second highest priority was to maintain dynamic balance by maintaining stability of all contacts. That can be written in the same form as the previous task by combining eq. (16) and (23). This task is always present, but only $\bar{\mathbf{Z}}$ changes its form, due to changes of contact configuration during the motion.

The next task imposed was that the CoM has to follow a predefined trajectory. This can be written as:

$$\mathbf{A}_3 = \mathbf{J}_C, \;\; \mathbf{b}_3 = \ddot{\mathbf{x}}_{3|des} - \dot{\mathbf{J}}_C \dot{\mathbf{q}} \tag{25}$$

where $\mathbf{J}_C$ represents a Jacobian matrix associated with CoM, and $\ddot{\mathbf{x}}_{3|des}$ represents the desired acceleration of the CoM. In case the robot needs to move one of its limbs, the following task must be included:

$$\mathbf{A}_4 = \mathbf{J}_P, \;\; \mathbf{b}_4 = \ddot{\mathbf{x}}_{4|des} - \dot{\mathbf{J}}_P \dot{\mathbf{q}} \tag{26}$$

where $\mathbf{J}_P$ is the Jacobian matrix associated with the limb the robot has to move and $\ddot{\mathbf{x}}_{4|des}$ is the desired acceleration of the limb. In case the robot needs to move more than one limb, Jacobian matrices for all limbs are concatenated in matrix $\mathbf{J}_P$. The same situation holds for $\ddot{\mathbf{x}}_{4|des}$. The fifth task the robot has to perform is to maintain its trunk vertical, which can be written as:

$$\mathbf{A}_5 = \mathbf{J}_{trR}, \; \mathbf{b}_5 = \ddot{\mathbf{x}}_{5|des} - \dot{\mathbf{J}}_{trR} \dot{\mathbf{q}}. \tag{27}$$

The Jacobian matrix associated with trunk rotation is $\mathbf{J}_{trR}$, while the desired trunk angular acceleration is $\ddot{\mathbf{x}}_{5|des}$. Desired acceleration is calculated as an output of simple a PD controller. The last task that the robot has to perform is to achieve a configuration which is as close as possible to the initial posture of the robot:

$$\mathbf{A}_5 = \left[\mathbf{0}_{N\times 6} \; \mathbf{I}_{N\times N}\right], \;\; \mathbf{b}_5 = \ddot{\mathbf{x}}_{5|des} \tag{28}$$

The number of joints of the robot is denoted as N, and position of the first 6 DoFs is not controlled since they represent the position of a free-floating base of the robot [25].

When tasks are defined, a generalized task prioritization framework's procedure for calculating control torques (Alg. 1) can be employed. Obtained motion is shown on Fig. 6, and it is clear that the robot was able to climb up the ramp by making additional contacts. When looking closely at the contacts, it is obvious that they remain stable throughout each of the several phases the robot passes through. In the first phase, the robot places its left foot on to the steep ramp (top-left). While doing so, the robot is in the single support phase. After that the robot grasps the handlebars (top-right). During that phase both robot's feet are in contact with the ground but with different surfaces. After grasping the handlebars the robot pulls itself up and places its right foot on top of the ramp (bottom-left), thus introducing internal loads. In this phase only the line contact is required between the front edge of the left foot and steep ramp. In order to move from surface contact in second phase to line contact in third phase, the dynamic balance matrix for that foot had to be recalculated. Contact between the hands of the robot and handlebars are modeled as a point contact (i.e. fingers of the hand are pushing against the forward-facing side of the handlebars). Finally, on the bottom right figure, the robot pulls itself forwards and places the left foot on the top of the ramp, while maintaining point contacts between the hands and handlebars and planar contact between the right foot and the top of the ramp. Internal load had to be introduced in the system in order to increase the normal contact force, thus avoiding sliding of the right foot. The amount of the internal load is not defined strictly as in [29], but the torques are calculated so that *Composite Dynamic Balance Matrix* constraint (16) is satisfied. The planning of the motion and feet placement is out of the scope of this paper.

On Fig. 7 the trajectory of CoM and centers of pressure for feet are shown. ZMP is never depicted, since contacts are never on the same plane, so the ZMP in its classical form cannot be defined. It can be clearly seen that when surface contact exists CoP of the foot is never on the edge, thus making contact potentially unstable. Also when line contact appears, CoP is on that line, but not at its ends, ensuring that the contacts remain stable. On the same figure it can be seen that the footprints did not move, meaning that sliding did not occur during the motion.

5.2 Walking sideways on inclined surface

In this scenario the robot should move sideways, while its feet are in contact with a surface angled at 45°. Although the friction coefficient is high ($\mu = 0.8$), the robot is unable to stand on the slope without additional contact between the hand and the horizontal surface, because the ground reaction force acting on the feet will always have a horizontal component which will tend to push the CoM towards the wall.

During the motion, the robot was always in contact with the environment with at least three of its limbs, while it was moving its fourth limb in the direction of the motion. To perform such a pattern, the robot had to fulfill several tasks, similar to the previous case. As before, the the highest priority was to maintain joint torques between the predefined saturation limits. The task of the second

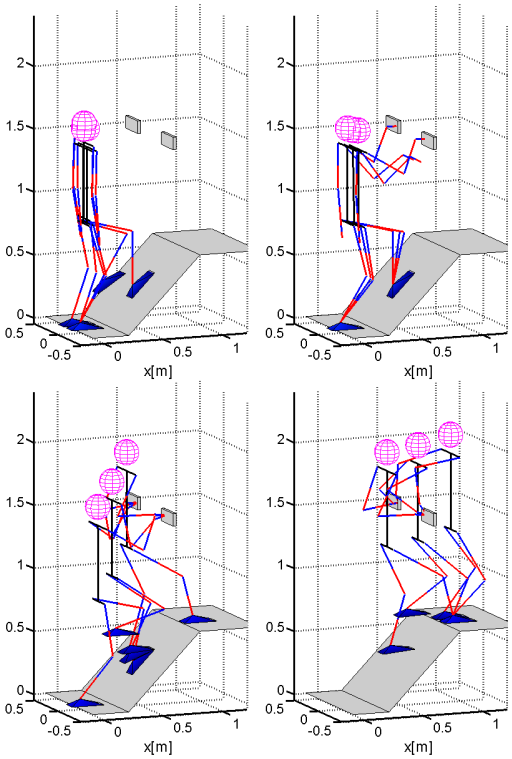


Fig. 6: Simulated motion when robot climbs up the steep ramp with the aid of the handlebars

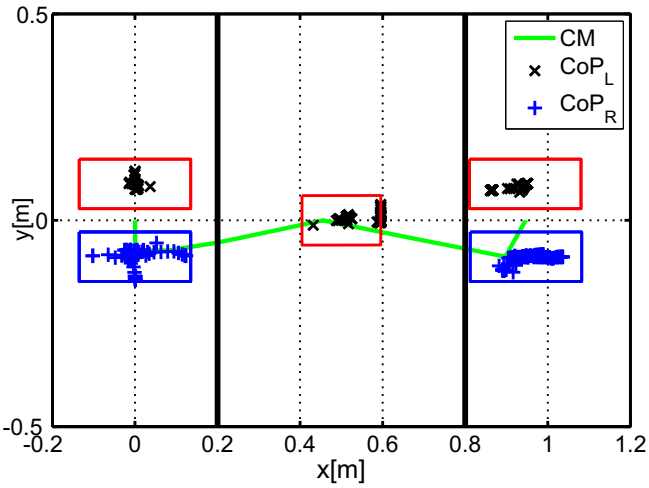


Fig. 7: Positions of CoM and centers of pressure (CoP) for feet

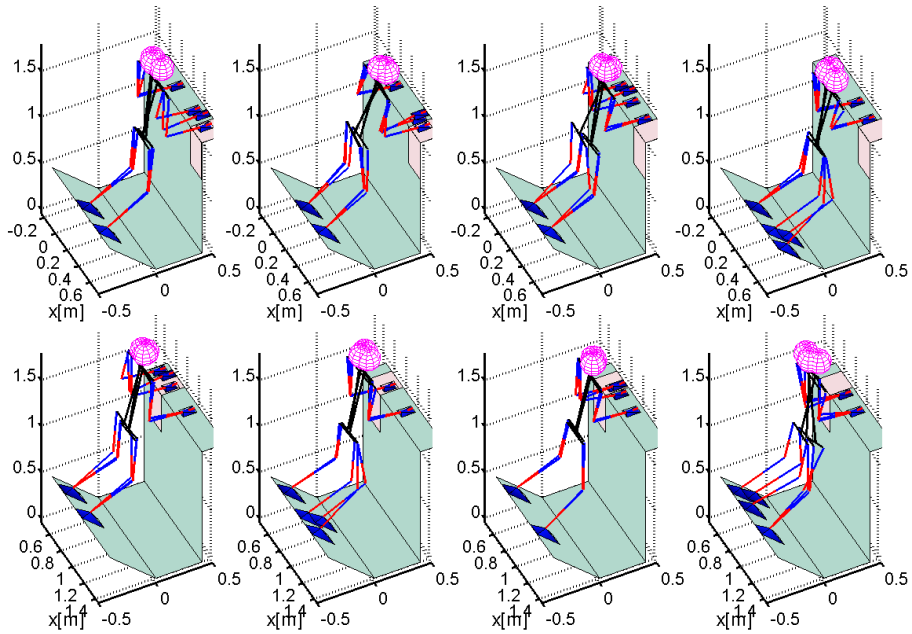
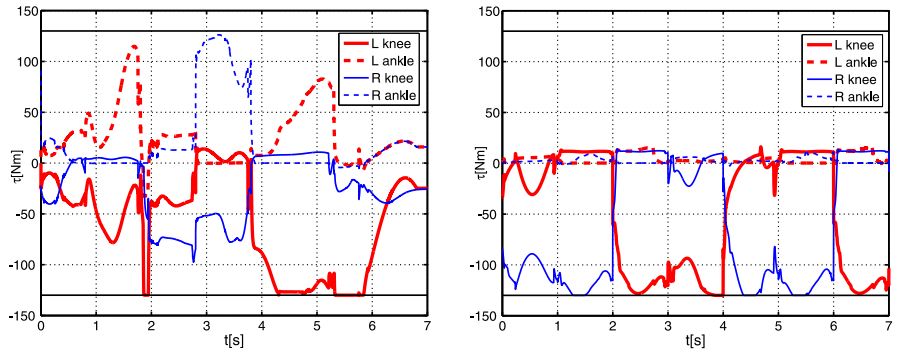


Fig. 8: Robot moving sideways. Each snapshot presents one phase of the motion when robot moves just one of its limbs. A pink ribbon represents the low friction area.

highest importance was to maintain all contacts stable. As mentioned earlier, it is inequality type task dependent on joint torques that can be obtained in required form (24) by combining eqns. (16) and (23). Matrix $\bar{\mathbf{Z}}$ changes over time because of the changes in configuration of the contacts. The next task that the robot had to perform was that CoM following a predefined trajectory. The task for repositioning one of the limbs had the second lowest priority. The last task that the robot had to perform was to maintain a configuration as close as possible to the initial posture of the robot.

When the robot moves sideways it repositions its limbs in the following order: right hand, left hand, right foot, left foot. The procedure for calculating torques described in Alg. 1 is employed, and the results of the simulation are shown in Fig. 8. The robot was able to move to the right with ease, until it puts the right hand on the low friction patch ($\mu = 0.2$ shown as a pink ribbon; first image on the top row). Before trying to reposition its next limb in order, the left hand, robot first checks if that would jeopardize the dynamic balance. The result shows that the robot would lose dynamic balance, since the hand on the low-friction patch would slide and the robot would fall. Because of that robot first repositions its right hand out of the low-friction patch. Only then robot is able to continue repositioning sequence, i.e. first to move left hand, then move right leg (third and fourth image in first row).

After several cycles, the robot places its left hand on the low-friction patch (first figure in bottom row). But in that case, the robot is able to move its right



(a) Climbing up (b) Walking sideways

Fig. 9: Torques at knee and ankle when robot is (a) climbing up the steep ramp and (b) walking sideways on inclined surface

leg forward with its left hand on a low-friction patch because the right hand takes most of the load. When robot wants to move its left foot forward, it is unable to do so, so it needs to remove the left hand from the low-friction patch first and then move the left foot forwards.

The scenarios presented in the paper show movements that would be hard for humans to perform and might require a lot of power. After looking at the simulation results, it can be seen that knee and ankle joints were closest to the saturation limits, preset to 130Nm (see Fig.9). It can be noted that torques in these joints are reaching the saturation limits several times. The task prioritization framework helps the robot perform the desired tasks while maintaining stable contact, even when motors reach torque saturation limits. The preset limit of 130Nm is three to four times the amount of torque needed for a regular walk of humans [6,38]. Although that might seem high, the research has shown that humans are able to achieve such high torques [1], so as the robots (for example state of the art humanoid robot Lola [21] can achieve such high torques). That means the proposed framework is generating physically realistic motions that can be performed by both humans modern humanoid robots.

6 Conclusion

In this paper we have proposed a novel criterion for dynamical balance called *Dynamic Balance Matrix*. It can be used for point, line and surface contacts of arbitrary shape. *Dynamic Balance Matrix* takes into account both separation of the surfaces in contact and sliding between them. It has been shown that this condition is both necessary and sufficient to provide contact stability. Correlation between the size of DBM and the shape of the foot, number of sides of friction cone and orientation of the cone has also been studied. An example was presented how required frictional forces and normal torque ('yaw friction') modulate the effective shape and size of the foot.

In addition, the proposed approach enables testing of dynamic balance in case the system has multiple spatially distributed contacts with the environment. In

such a case *Composite Dynamic Balance Matrix* should be used. This approach can also be used to verify the dynamic balance of planned motion. If it is infeasible, motion can be replanned with additional contacts, until it becomes feasible. One such example was used to illustrate the proposed approach and whole-body motion synthesis using generalized task-prioritization framework. Promising results were obtained after interfacing qpOASES which lowered the time of calculation of desired torques to $5ms$, which gives assurance that this method could be used for on-line whole-body motion synthesis.

References

1. Aagaard, P., Simonsen, E.B., Trolle, M., Bangsbo, J., Klausen, K.: Moment and power generation during maximal knee extensions performed at low and high speeds. European Journal of Applied Physiology and Occupational Physiology **69**(5), 376–381 (1994). DOI 10.1007/BF00865398. URL `http://dx.doi.org/10.1007/BF00865398`
2. Barber, C.B., Dobkin, D.P., Huhdanpaa, H.: The quickhull algorithm for convex hulls. ACM Trans. Math. Softw. **22**(4), 469–483 (1996). DOI 10.1145/235815.235821. URL `http://doi.acm.org/10.1145/235815.235821`
3. Caron, S., Pham, Q.C., Nakamura, Y.: Leveraging cone double description for multi-contact stability of humanoids with applications to statics and dynamics. In: Robotics: Science and System (2015)
4. Caron, S., Pham, Q.C., Nakamura, Y.: Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench for rectangular support areas. In: Robotics and Automation (ICRA), 2015 IEEE International Conference on. IEEE (2015)
5. Dai, H., Valenzuela, A., Tedrake, R.: Whole-body motion planning with centroidal dynamics and full kinematics. In: Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on, pp. 295–302 (2014). DOI 10.1109/HUMANOIDS.2014.7041375
6. DeVita, P., Hortobgyi, T.: Obesity is not associated with increased knee joint torque and power during level walking. Journal of Biomechanics **36**(9), 1355 – 1362 (2003). DOI http://dx.doi.org/10.1016/S0021-9290(03)00119-2. URL `http://www.sciencedirect.com/science/article/pii/S0021929003001192`
7. Escande, A., Kheddar, A., Miossec, S.: Planning contact points for humanoid robots. Robotics and Autonomous Systems **61**(5), 428 – 442 (2013). DOI http://dx.doi.org/10.1016/j.robot.2013.01.008. URL `http://www.sciencedirect.com/science/article/pii/S0921889013000213`
8. Fallon, M.F., Marion, P., Deits, R., Whelan, T., Antone, M., McDonald, J., Tedrake, R.: Continuous humanoid locomotion over uneven terrain using stereo fusion. In: 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), pp. 881–888 (2015). DOI 10.1109/HUMANOIDS.2015.7363465
9. Faraji, S., Pouya, S., Ijspeert, A.: Robust and Agile 3D Biped Walking With Steering Capability Using a Footstep Predictive Approach. In: Robotics Science and Systems (RSS) (2014). URL `http://www.roboticsproceedings.org/rss10/p28.html`
10. Ferreau, H., Kirches, C., Potschka, A., Bock, H., Diehl, M.: qpOASES: A parametric active-set algorithm for quadratic programming. Mathematical Programming Computation **6**(4), 327–363 (2014)
11. Fukuda, K., Prodon Alain", e.M., Euler, R., Manoussakis, I.: Combinatorics and Computer Science: 8th Franco-Japanese and 4th Franco-Chinese Conference Brest, France, July 3–5, 1995 Selected Papers, chap. Double description method revisited, pp. 91–111. Springer Berlin Heidelberg, Berlin, Heidelberg (1996). DOI 10.1007/3-540-61576-8_77
12. Goyal, S., Pinson, E.N., Sinden, F.W.: Simulation of dynamics of interacting rigid bodies including friction i: General problem and contact model. Engineering with computers **10**(3), 162–174 (1994)
13. Goyal, S., Pinson, E.N., Sinden, F.W.: Simulation of dynamics of interacting rigid bodies including friction ii: Software system design and implementation. Engineering with computers **10**(3), 175–195 (1994)
14. Harada, K., Hirukawa, H., Kanehiro, F., Fujiwara, K., Kaneko, K., Kajita, S., Nakamura, M.: Dynamical balance of a humanoid robot grasping an environment. In: Intelligent

Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, vol. 2, pp. 1167–1173. IEEE (2004)
15. Harada, K., Kajita, S., Kaneko, K., Hirukawa, H.: Dynamics and balance of a humanoid robot during manipulation tasks. Robotics, IEEE Transactions on **22**(3), 568–575 (2006)
16. Herzog, A., Rotella, N., Mason, S., Grimminger, F., Schaal, S., Righetti, L.: Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid. Autonomous Robots **40**(3), 473–491 (2015). DOI 10.1007/s10514-015-9476-6
17. Herzog, A., Rotella, N., Schaal, S., Righetti, L.: Trajectory generation for multi-contact momentum control. In: Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on, pp. 874–880 (2015). DOI 10.1109/HUMANOIDS.2015.7363464
18. Hirukawa, H., Hattori, S., Harada, K., Kajita, S., Kaneko, K., Kanehiro, F., Fujiwara, K., Morisawa, M.: A universal stability criterion of the foot contact of legged robots-adios zmp. In: Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, pp. 1976–1983. IEEE (2006)
19. Kuindersma, S., Permenter, F., Tedrake, R.: An efficiently solvable quadratic program for stabilizing dynamic locomotion. In: Robotics and Automation (ICRA), 2014 IEEE International Conference on, pp. 2589–2594 (2014). DOI 10.1109/ICRA.2014.6907230
20. Lengagne, S., Vaillant, J., Yoshida, E., Kheddar, A.: Generation of whole-body optimal dynamic multi-contact motions. The International Journal of Robotics Research **32**(9-10), 1104–1119 (2013)
21. Lohmeier, S., Buschmann, T., Ulbrich, H., Pfeiffer, F.: Humanoid Robot LOLA — Research Platform for High-SpeedWalking, pp. 221–230. Springer Netherlands, Dordrecht (2009). DOI 10.1007/978-1-4020-9438-5_22. URL `http://dx.doi.org/10.1007/978-1-4020-9438-5_22`
22. Mattingley, J., Boyd, S.: Cvxgen: a code generator for embedded convex optimization. Optimization and Engineering **13**(1), 1–27 (2011). DOI 10.1007/s11081-011-9176-9. URL `http://dx.doi.org/10.1007/s11081-011-9176-9`
23. Nikolić, M., Borovac, B., Raković, M.: Walking on slippery surfaces: Generalized task-prioritization framework approach. In: Advances on Theory and Practice of Robots and Manipulators: Proceedings of Romansy 2014 XX CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators, pp. 189–196. Springer (2014)
24. Nikolić, M., Borovac, B., Raković, M., Savić, S.: A further generalization of task-oriented control through tasks prioritization. International Journal of Humanoid Robotics **10**(03) (2013)
25. Nikolic, M., Savic, S., Borovac, B., Rakovic, M.: Task prioritization framework for kinesthetic teaching of a free-standing humanoid robot. In: Intelligent Systems and Informatics (SISY), 2015 IEEE 13th International Symposium on, pp. 241–246 (2015). DOI 10.1109/SISY.2015.7325387
26. Pang, J.S., Trinkle, J.: Stability characterizations of rigid body contact problems with coulomb friction. ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift fr Angewandte Mathematik und Mechanik **80**(10), 643–663 (2000). DOI 10.1002/1521-4001(200010)80:10⟨643::AID-ZAMM643⟩3.0.CO;2-E. URL `http://dx.doi.org/10.1002/1521-4001(200010)80:10<643::AID-ZAMM643>3.0.CO;2-E`
27. Sentis, L., Khatib, O.: Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. International Journal of Humanoid Robotics **2**(04), 505–518 (2005)
28. Sentis, L., Khatib, O.: A whole-body control framework for humanoids operating in human environments. In: Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on, pp. 2641–2648 (2006). DOI 10.1109/ROBOT.2006.1642100
29. Sentis, L., Park, J., Khatib, O.: Compliant control of multicontact and center-of-mass behaviors in humanoid robots. Robotics, IEEE Transactions on **26**(3), 483–501 (2010)
30. Takao, S., Yokokohji, Y., Yoshikawa, T.: Fsw (feasible solution of wrench) for multi-legged robots. In: Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on, vol. 3, pp. 3815–3820. IEEE (2003)
31. Udwadia, F.E., Kalaba, R.E.: Analytical Dynamics: A New Approach. Cambridge University Press, Cambridge, UK (1996)
32. Vukobratović, M., Borovac, B.: Zero-moment pointthirty five years of its life. International Journal of Humanoid Robotics **1**(01), 157–173 (2004)
33. Vukobratović, M., Juricic, D.: Contribution to the synthesis of biped gait. Biomedical Engineering, IEEE Transactions on (1), 1–6 (1969)
34. Vukobratović, M., Potkonjak, V., Babković, K., Borovac, B.: Simulation model of general human and humanoid motion. Multibody System Dynamics **17**(1), 71–96 (2007)

35. Vukobratović, M., Stepanenko, J.: On the stability of anthropomorphic systems. Mathematical Biosciences **15**(1), 1–37 (1972)
36. Wensing, P.M., Orin, D.E.: Generation of dynamic humanoid behaviors through task-space control with conic optimization. In: Robotics and Automation (ICRA), 2013 IEEE International Conference on, pp. 3103–3109 (2013). DOI 10.1109/ICRA.2013.6631008
37. Whelan, T., Kaess, M., Johannsson, H., Fallon, M., Leonard, J.J., McDonald, J.: Real-time large-scale dense rgb-d slam with volumetric fusion. The International Journal of Robotics Research **34**(4-5), 598–626 (2015). DOI 10.1177/0278364914551008. URL `http://dx.doi.org/10.1177/0278364914551008`
38. Winter, D.A.: Biomechanics and Motor Control of Human Movement, 3 edn. Wiley (2004)