Fast Whole-Body Motion Control of Humanoid Robots with Inertia Constraints

Grzegorz Ficht and Sven Behnke

Abstract—We introduce a new, analytical method for generating whole-body motions for humanoid robots, which approximate the desired Composite Rigid Body (CRB) inertia. Our approach uses a reduced five mass model, where four of the masses are attributed to the limbs and one is used for the trunk. This compact formulation allows for finding an analytical solution that combines the kinematics with mass distribution and inertial properties of a humanoid robot. The positioning of the masses in Cartesian space is then directly used to obtain joint angles with relations based on simple geometry. Motions are achieved through the time evolution of poses generated through the desired foot positioning and CRB inertia properties. As a result, we achieve short computation times in the order of tens of microseconds. This makes the method suited for applications with limited computation resources, or leaving them to be spent on higher-layer tasks such as model predictive control. The approach is evaluated by performing a dynamic kicking motion with an igus® Humanoid Open Platform robot.

I. INTRODUCTION

Control of humanoid robots is a challenging task, as they belong to a class of high-dimensional, underactuated systems. Full mathematical formulations containing all of the precise physical details about the system are highly complex. As every moving body part contributes to the overall dynamic behaviour, motion planners are required to traverse a broad search space, resulting in long solve times. As a solution, several simplified models have been used, which approximate the general dynamics of the system.

Single mass pendulum models are quite popular in this regard, as they allow to characterise the linear momentum of the system through a trajectory of the Center of Mass (CoM). A linear simplification of the inverted pendulum [1] has been widely used in bipedal walking [2]. Through such simple models, even comprehensive and efficient control strategies can be achieved [3]. The beneficial simplicity is accompanied with limitations of not including the angular momentum and a constant CoM height. Three mass models [4][5] were used to model leg swing dynamics and accordingly alter the Zero Moment Point (ZMP) [6] for stable bipedal walking. Once the dynamic trajectories have been computed, the joint states are then usually computed using numerical inverse kinematics to satisfy the desired CoM. A generalisation for the centroidal dynamics [7] was proposed by Lee et al. [8] with the Reaction Mass Pendulum (RMP), where six equal masses sliding on three actuated rails shape the total inertia of the robot. The masses are abstract and do not relate to the joint angles in a direct way. As a result, the mapping between inertia and whole-body pose is not unique.

More general whole-body controllers that address a number of criteria have also been developed. One of these—the Resolved Momentum Control framework—unifies control over linear and angular momenta [9] and generated naturally looking kicking and walking motions. To solve broader and less-structured problems than biped locomotion, task-based optimisation methods are commonly employed. Through their usage, various multi-task [10] [11] problems can be solved. Depending on the constraints and details included in the model, the solver can take up to hours of computation [12]. Recent works have been targeted at reducing solve times through simplifying the dynamics [13] or through their more efficient computation [14].

In our previous work [15], we have shown that whole-body poses that satisfy a given CoM placement can be computed using a five mass model and relatively simple geometry with a triangle mass distribution. At the same time, Arreguit et al. [16] used a five mass model with stretchable limbs for simplified dynamics in motion planning tasks to achieve solve times in the order of milliseconds. The five mass model efficiently encodes the combination of limb swing dynamics and their kinematics (Fig. 1). As we show in this work, it also provides insight in determining pose reachability constraints and allows for generating whole-body motions analytically in microseconds. The contribution of this paper is a novel, efficient method for controlling humanoid robots through setting a combination of whole-body poses that satisfy a given CoM position and rotational inertia. A video is available1.

II. REDUCED FIVE MASS MODEL DESCRIPTION

We describe a humanoid robot as a tree-structured rigid body consisting of five point-masses with a non-uniform mass distribution. A 6 degree of freedom (DoF) floating base

1http://ais.uni-bonn.de/videos/ICRA_2020_Ficht
frame $\Sigma_B$ is attached to the pelvis of the torso, between the hip joint origins $\mathbf{b}_m$. We consider the head a part of the torso, and place this combined mass $m_t$ at a configurable offset from the base. This leaves the head unconstrained, which is desirable in vision systems. The remaining four links are attached to the torso and represent the (left and right) leg $m_{ll}, m_{rl}$ and arm $m_{la}, m_{ra}$, which originate at the hip and shoulder joints respectively. The centroidal properties of the system, i.e. CoM position $\mathbf{m}_c$ and orientation of the principal axes of inertia $\mathbf{R}_l$ are combined in frame $\Sigma_C$. All movable joints such as hips, knees, ankles, shoulders, elbows and wrists possess their own frame. Furthermore, two foot frames $\Sigma_{FL}, \Sigma_{FR}$ are placed at $\mathbf{f}_l, \mathbf{f}_r$ and represent the centers of the foot polygon. For the convenience of further calculations, all position vectors are expressed in the CoM frame, i.e. $\mathbf{m}_c = (0,0,0)$, while orientations follow a right-handed coordinate system.

The limb mass position is tied-in with the kinematics through a triangle approximation. Two sides of the triangle are of known, constant length $(c,a)$ and represent the actual upper and lower leg and arm links of the robot. The third side corresponds with the variable limb extension $b$, dependent on the knee or elbow bending angle. In a limb, two mass distribution parameters are used to determine the relative position of the mass in the triangle with respect to the origin:

$$P = (p_s, p_l) \in [0,1].$$

The side distribution parameter $p_s$ is used to section the lower link of the limb. The center of mass is located along the vector formed between this intersection point and the limb origin. The length distribution parameter $p_l$ represents the ratio between the CoM distance to the length of this vector. A triangle with uniform density, for example, has $p_s = \frac{1}{2}$, $p_l = \frac{2}{3}$, with the vector between the origin and CoM being a median. By assigning $P$ values for each limb, a one-to-one mapping is formed through which joint values can be obtained from mass positioning, and vice versa. Additionally, inverse distribution parameters $P_l = (p_{sl}, p_{ll})$ can be calculated through simple intersections. These can then be used to find the configuration of a limb depending on the mass distance from the wrist or ankle. The connection between mass and limb movement is depicted on Fig. 2. Although the mass of the feet and hands is included in the total limb mass, it is assumed that their orientation does not influence the final mass placement in the triangle. This is due to their limited range of motion and low weight relative to the rest of the limb. As shown in previous work [15], such a relationship between the mass distribution and limb kinematics is precise, given that $P$ was determined with sufficient accuracy.

This five link formulation is more complex than typically used single point-mass models, however it possesses beneficial properties which simpler models are unable to capture. The main advantage is the ability to describe the robot as an equivalent Composite Rigid Body (CRB) with varying rotational inertia. This allows to characterise the centroidal dynamic properties of the system, and control the linear and angular momentum. In comparison to the already mentioned Reaction Mass Pendulum [8], the masses are not abstract and have a direct connection to the physical system. Not only do they describe the non-uniform mass distribution, but also control the kido-dynamic properties. Meaning that a change in the positioning of a single mass, affects the CRB inertia and the limb configuration in a clear and unique way.

### III. Analytic Whole-Body Control

Using the model described in the previous section, we propose a novel method for whole-body control. We capitalise on the relationships of mass distribution and kinematics, and derive analytic solutions based on geometry. For the pose generation process, the desired foot placement and inertial properties are used. This input, combined with the physical system properties is then used to compute target joint values. The complete process flow of generating a whole-body pose is presented in Fig. 3.

#### A. Constraint inclusion

A number of constraints are considered which act as the system input. The highest priority is given to CoM placement, which is constrained by the foot positions. The inertial properties of the multibody are considered to be of lower priority, where the inertia orientation takes precedence over the principal moments in some cases. As we show in Section III-C, foot placement is the main constraint defining factor with respect to both the CoM and inertia. Lastly, when a pose has been calculated that respects the possible limits, an optional constraint on the trunk orientation can be
The masses connected with a massless rod of length \( l \) achieve the set distance in terms of desired inertia:

\[
I = \frac{m_l}{m_l + m_u}, \quad l_u = \frac{m_l}{m_l + m_u}.
\]

Finally, we compute the desired lower and upper body mass placement \( m_l, m_u \), which realises the set tilt inertia:

\[
m_l = R_1 \begin{bmatrix} 0 \\ 0 \end{bmatrix} (-l_l), \quad m_u = R_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} l_u.
\]

### C. Foot placement influence

Due to kinematic constraints, a whole-body pose that satisfies all of the constraints might not always be attainable. A large part of that is dependent on the placement of the feet. Given the desired foot placement \( \Sigma_{FL}, \Sigma_{FR} \), the left and right ankle position \( a_l, a_r \) can be computed as:

\[
a = f_e + R_F, \ OR_e,
\]

where \( R_F \in \text{SO}(3) \) is the rotation matrix with the foot orientation, and \( OR_e \) is the position offset from the center of that foot polygon \( f_e \) to its ankle \( a_e \). A maximum CoM extension is reached, when all of the masses are located in a single, opposite direction to the feet, with respect to kinematic constraints. For a human or an anthropomorphic humanoid robot, it is an upright standing pose with fully stretched legs and arms extended upwards. Naturally, the hip midpoint \( h_m \) is also furthest away from the feet. In terms of the dumbbell model from Section III-B, at this full hip extension the maximum distance between the ankles \( a_l, a_r \) and lower body mass \( m_l \) is achieved. The maximum hip midpoint \( h_m \) and \( m_l \) range is deduced as in Fig. 4. From the ankles, fully extended legs connect at a distance defined by the hip width \( h_w \). A virtual ankle midpoint \( a_m \) is placed on the line connecting both ankles, which are separated by distance \( s \). Additionally, \( a_h \) representing the aggregate heading of the feet \( x \)-axis, is placed at an offset from \( a_m \). The lower body mass then lies inside the intersection of two spheres placed at the ankles, each with a radius of \( r_l \) defining the masses distance from the legs:

\[
\| a - m \| \leq r_l,
\]

where the value of \( r_l \) is calculated simply as:

\[
r_l = \sqrt{\left( \frac{s}{2} \right)^2 + \| a_m - m \|_{\text{max}}^2}.
\]

The distance between the ankle midpoint and lower body mass in this case is found through a ratio of mass placement in terms of hip extension (\( \| h_m - a_m \| \)) and a chosen leg:

\[
\| a - a_m \|_{\text{max}} = \| a_m - a \|_{\text{max}} = \| a_m - h_m \|_{\text{max}}
\]

where the individual lengths for a chosen leg are:

\[
\| a - h \|_{\text{max}} = c + a, \\
\| a_m - a \|_{\text{max}} = c + a - p_l(c + p) a, \\
\| a_m - h_m \|_{\text{max}} = \sqrt{(c + a)^2 - \left( \frac{s - h_m}{2} \right)^2}.
\]
It can be observed, that a maximum hip extension is achieved when the separation between ankles is equal to the hip width. Both increasing and decreasing \( s \), results in a decreased range of motion for \( h_m \) and—as a result—\( m_l \). Given a set CoM position, the orientation and amplitude of the tilting inertia is limited by the sphere intersection (10), as shown in Fig. 5a.

In case the set inertia results in \( m_l \) being outside of this area, we find a solution close to the desired one and recompute the inertial parameters. Ideally, it is possible to preserve the value of \( I_z \) and tilt angle, only altering the tilt angle to achieve the closest point on the surface of the sphere intersection. This however, leads to discontinuous solutions for larger angles, where the closest point might be opposite the intended one. Instead, an intersection of a ray from the hip midpoint which satisfies the lower mass placement of foot and arm mass, with the arms pointing towards the virtual leg plane, defined by three points: \( a_m, a_h \) and \( m_l \). As all of the sides of the triangle formed by \( a_m, k, \) and \( p \) are of known length, the angle of rotation \( \phi \) is calculated as the adjacent angle to \( \theta \) at \( p \):

\[
\phi = \pi - \theta
\]

\[
\theta = \acos \left( \frac{p_i^v \cdot (a^v)}{2(a^v \cdot p_i^v) ||a_m - m_l||} \right)
\]

A visual representation of this solution is shown in Fig. 6. Having both \( h_m \) and \( m_{al} \), it is necessary for the distance between them to be within range of \( d_u \). In such a case, the pose that satisfies the CoM and the tilting inertia can be reached and is considered valid and the following steps can be omitted. Otherwise, the tilting inertia requires further adjustment in either the orientation \( R_1 \) or amplitude \( l_z \). Priority is given to keep the set orientation, as having control over it through time still allows for generating angular momentum around the CoM. By setting \( d_u \) to a value in the possible range \( d_s \), leaves (14) then as a function of only \( l_f \). Unfortunately this is an octic equation, for which there is no general solution. However, the influence of \( l_f \) on the placement of \( m_1, m_u \) and \( h_m \) can be observed from (8) and (16). The hip midpoint \( h_m \) amplifies the distance between \( m_1 \) and \( m_u \) directly set by \( l_f \). Given that the CoM placement is limited by the fully extended pose (including the influence of foot separation \( s \)), there exists a \( l_f \) value that satisfies the desired \( d_u^* \):

\[
f(l_f) = d_u - d_s = ||h_m - m_u|| - d_s
\]

which can be found iteratively. Although the derivative of the function can be calculated for gradient-based optimisation, it is computationally unfavorable as it results in almost 300 operations on over 40 subexpressions, made up from 2 to 15 operations each. Instead, we perform a search between two \( l_f \) values which produce opposite sign results on (19), as shown in Fig. 7.
The evaluation of the initial mass placement (14) is used to obtain $d_s$. Either the pose is valid and $\mathbf{m}_u$ can be reached, or the hip is too close ($d_s = d_{\text{min}}$) or too far ($d_s = d_{\text{max}}$) from the upper mass. As seen on Fig. 7, three points are evaluated to assess the range of $l_I$ for the search algorithm. Points $\mathbf{m}_1$, $\mathbf{m}_2$ lie on the intersection of the tilting inertia and the lower mass placement region. If the set CoM placement is at the limit, where a fully extended pose is required, both $\mathbf{m}_1$ and $\mathbf{m}_2$ will be 'too far' from the solution, which requires to alter not only $l_I$, but $\mathbf{R}_1$ as well. To assure that the CoM constraint is fulfilled, $\mathbf{m}_3$ is calculated as the intersection of the ray $\mathbf{a}_u \mathbf{m}_c$ ($l_I = 0$) and the allowed lower mass region. We then compute (19) for all three points, two of which are used as the $l_I$ interval. The search is then performed with one of the various root-finding algorithms, until a sufficient accuracy has been reached e.g. bisection, regula falsi and their variants. Searching through the range of $\mathbf{m}_2$ and $\mathbf{m}_3$ (as compared to $\mathbf{m}_1$ and $\mathbf{m}_2$) is slightly slower as it requires an additional recomputation of the intersection point and corresponding inertia orientation and mass placement at every step.

E. Complete mass placement and CRB inertia

With a reachable pose found that produces a given tilt inertia, the base frame position $\mathbf{h}_u$ is set. The base frame z-axis direction $z$ is then computed to bring the torso mass $\mathbf{m}_u$ as close to the set upper mass $\mathbf{m}_u$ and its yaw angle $\psi_t$ is aligned to the desired inertia yaw $\psi_t$. This completes the base frame $\Sigma_B$ definition. Additionally, if according to (14) $d_s$ is far from the limits, an additional trunk tilt orientation constraint $\theta_s$ can be employed. Then, the final orientation is interpolated between $\theta_s$ and $\theta_u$, where the maximum interpolation is limited by the arm reach: $d_{\text{max}} - d_s$.

Using the base frame, we locate the hip $\mathbf{h}_l$, $\mathbf{h}_r$ and shoulder $\mathbf{s}_l$, $\mathbf{s}_r$ origins. The final leg configurations are then computed with the triangle approximation [15] based on the set foot frames $\Sigma_FL$, $\Sigma_FR$ and hip origins, which produce the leg mass placement $\mathbf{m}_{u_l}$, $\mathbf{m}_{r_l}$. As these masses are off the tilt axis (separated by distance $s_l$), they contribute to the $I_{xx}$ and $I_{yy}$ components of the principal moments. This is done through a combination of the lower body yaw inertia $I_l$ and the lower body yaw angle $\psi_l$, formed between the line connecting the two point masses and the tilt inertia y-axis. A similar two-mass distribution ($\mathbf{m}_{u_l}$, $\mathbf{m}_{r_u}$) for the upper body ($s_u$, $\psi_u$, $I_u$) can then be used to achieve the given inertia yaw orientation, to generate or counteract yaw momentum. The distance and angle of the upper body mass particles around $\mathbf{m}_{u_r}$ can be calculated through the difference in inertia:

$$s_u = \sqrt{\left(\frac{I_u - I_l}{m_u + m_{ra}}\right)\left(\frac{m_{ra}}{m_u + m_{ra}}\right)\left(\frac{m_{ra}}{m_{ra}}\right)}$$

$$\psi_u = \frac{(m_u + m_t)\psi_t - m\psi_l}{m_u}$$

where $I_u$ is the total yaw inertia around the inertia z-axis. Ideally, the ratio between the angles should be determined by the yaw inertia. We use the mass distribution instead—to avoid having a zero or negative value in the denominator—when the legs provide the complete required angular momentum ($I_u - I_l \leq 0$, $s_u = 0$). To avoid self-collisions, the arm placement $\mathbf{m}_{la}$, $\mathbf{m}_{ra}$ is computed as to keep a minimum distance to the trunk with the arm yaw angle $\psi_u$. Otherwise, the mass distribution is used:

$$\mathbf{m}_{la} = \frac{\mathbf{m}_{la}(m_r + m_{ra}) - \mathbf{m}_l(m_u)}{m_{ra}}$$

$$\mathbf{m}_{ra} = \frac{\mathbf{m}_{u_r}(m_r + m_{ra}) - \mathbf{m}_l(m_u)}{m_{ra}}$$

A visual representation of the yaw inertia components and arm mass placement is shown on Fig. 8.

IV. EXPERIMENTAL RESULTS

Verification of the proposed approach was performed on a 90 cm tall, igus® Humanoid Open Platform robot [20] possessing 20 position-controlled joints. The motions were generated using the robots on-board computer.

A. Modelling error

First, we verify the validity and precision of the method by generating whole-body poses based on varying set inertial properties using the assumed model. The results are compared to a precise model with all masses and inertias, acting as ground truth. As seen on Fig. 9, the CoM positioning is precise, with a mean error and standard deviation of 1.5 mm in spite of large variations in the pose. For typical upright-scenarios, the CoM error is almost non-existent. The model also accurately captures the inertial properties, with small deviations in the principal moments ($I_{xx}$, $I_{yy}$, $I_{zz}$). The largest errors occur in the orientation $\mathbf{R}_1$. As the model consists of only point masses, the inertia products of the trunk around...
the trunk yaw angle $\psi$ in the roll orientation influence the yaw and vice versa. An upper body movement that counterbalances the momentum is set to zero, which results in producing vivid and natural support. During the kick, the centroidal angular momentum to perform the kick, after which the robot returns to double shifted above the stance foot, which enables the other foot designed momentum and foot frame setpoints. The CoM is computed and performed on the robot on-line, using pre-kick as seen on Fig. 10. The joint states trajectories were B. Kicking motion

We test the motion generation by performing a dynamic kick as seen on Fig. 10. The joint states trajectories were computed and performed on the robot on-line, using pre-designed momentum and foot frame setpoints. The CoM is shifted above the stance foot, which enables the other foot to perform the kick, after which the robot returns to double support. During the kick, the centroidal angular momentum is set to zero, which results in producing vivid and natural upper body movement that counterbalances the momentum generated by the legs. Visually, the result is similar to that of the one generated by the Resolved Momentum Controller [9].

C. Computation time

A common point in whole-body control is the time necessary to generate a frame of motion. We prepared three scenarios, in which the reachability of the pose influences the computation. The C++ implementation of the method was executed on a single core of an Intel i7-4710MQ processor set to its base frequency of 2.5 GHz and timed over 10000 executions. The achieved results are shown in Table I. On average, the algorithm requires only 33 $\mu$s to complete, which is attributed to the procedural approach using simple geometric relations. The method is the fastest when the set values can be met as it does not require to recompute a valid pose. For the search, regula falsi was used, which was able to achieve an error on (19) below 0.1 mm within 2 to 3 iterations. The quick convergence hints that (19) is mostly linear in the search interval. At the used update rate of 100 Hz, approximately 300 motion frames can be evaluated before a single one is executed, which opens up possibilities for on-line model predictive control. Moreover, it is expected that a microcontroller implementation of the method running with a 100 Hz loop is feasible, which is of importance in low-cost robotics.

V. Conclusions

A novel approach to whole-body control of humanoid robots has been presented that uses a simplified non-uniform mass distribution model to produce desired inertial properties. The method demonstrates that control over several physical quantities of highly-dimensional and complex humanoid robots can be achieved with relatively simple, analytic means. As a result, fast computation times are achieved, which leaves room for integration of higher-layer control methods. As the proposed solution generates motions considering inertia constraints, it becomes a useful component in locomotion control schemes, where both linear and angular momentum are of importance. In this regard, it significantly outperforms our prior, purely CoM-based approach [15]. The whole-body control has been experimentally verified and produced a naturally looking kick with a real, physical robot.

The method can be transferred to other anthropomorphic humanoids, given that the mass distribution is properly identified. This applies to biped robots as well, although with limitations in terms of the desired inertial properties. Other, more general task-optimising approaches can also benefit from our results, as warm starting the solver with a quickly generated pose, can lead to faster convergence rates. In future work, we want to combine the whole-body controller with feedback control methods for balance in locomotion tasks.

| TABLE I |
| --- |
| CONSTRAINTS MET | MEAN / SD ($\mu$s) |
| CoM + $R_1 + I_z$ | 26.6475 / 3.6815 |
| CoM + $R_1$ | 32.5346 / 7.3104 |
| CoM | 40.6502 / 7.6436 |

Fig. 9. Modelling error of the approach compared to a complete, precise model. From the top: Principal moments of inertia, inertia orientation, and evolution of poses performed by the robot over time. Dashed lines shows set values, while read values are solid. Inertia x-axis ($I_{xx}$, roll) values are red, y-axis ($I_{yy}$, pitch) green and z-axis ($I_{zz}$, yaw) blue. CoM height is shown in black. The inertia is visualised as an equimomental ellipsoid (transparent red) placed at the CoM, reflecting the mass distribution along an axis.

the 'spine' ($h_m - m_1$) are not included. Therefore, poses with a sagitally non-symmetric trunk orientation have an influence on the inertia, although the CoM placement is still met. This is visible in situations with ($0 - 2.4$s) and without ($2.4 - 5.5$s) a set yaw orientation, where changes in the roll orientation influence the yaw and vice versa. Currently, the trunk yaw angle $\psi_t$ is aligned with the set inertia yaw $\psi_I$, so as to force the arms to compensate the leg movement. A revision of this strategy might provide more accurate results with respect to the inertia orientation. Although influenced by the trunk movement, the tilting inertia orientation is generally achieved.

B. Kicking motion

We test the motion generation by performing a dynamic kick as seen on Fig. 10. The joint states trajectories were computed and performed on the robot on-line, using pre-designed momentum and foot frame setpoints. The CoM is shifted above the stance foot, which enables the other foot to perform the kick, after which the robot returns to double support. During the kick, the centroidal angular momentum is set to zero, which results in producing vivid and natural upper body movement that counterbalances the momentum generated by the legs. Visually, the result is similar to that of
REFERENCES

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

[2] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, “Biped walking pattern generation by using preview control of zero-moment point,” in Robotics and Automation, 2003. Proceedings. ICRA’03. IEEE International Conference on, vol. 2, 2003, pp. 1620–1626.

[3] M. Missura and S. Behnke, “Omnidirectional capture steps for bipedal walking,” in Humanoid Robots (Humanoids), 2013 13th IEEE-RAS International Conference on, 2013, pp. 14–20.

[4] T. Takenaka, T. Matsumoto, and T. Yoshikane, “Real time motion generation and control for biped robot-1 st report: Walking gait pattern generation,” in Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on, 2009, pp. 1084–1091.

[5] S. Shimmyo, T. Sato, and K. Ohnishi, “Biped walking pattern generation by using preview control based on three-mass model,” IEEE transactions on industrial electronics, pp. 5137–5147, 2012.

[6] M. Vukobratović and B. Borovac, “Zero-Moment Point - Thirty Five Years of its Life,” International Journal of Humanoid Robotics, vol. 1, no. 01, pp. 157–173, 2004.

[7] D. E. Orin, A. Goswami, and S.-H. Lee, “Centroidal dynamics of a humanoid robot,” Autonomous Robots, vol. 35, no. 2-3, pp. 161–170, 2013.

[8] S.-H. Lee and A. Goswami, “Reaction Mass Pendulum (RMP): An explicit model for centroidal angular momentum of humanoid robots,” in Robotics and Automation (ICRA), IEEE International Conference on, 2007, pp. 4667–4672.

[9] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, “Resolved Momentum Control: Humanoid Motion Planning based on the Linear and Angular Momentum,” in Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on, vol. 2, 2003, pp. 1644–1650.

[10] L. Sentis and O. Khatib, “Synthesis of Whole-Body Behaviors through Hierarchical Control of Behavioral Primitives,” International Journal of Humanoid Robotics, vol. 2, no. 04, pp. 505–518, 2005.

[11] A. De Prete, F. Romano, L. Natale, G. Metta, G. Sandini, and F. Nori, “Prioritized Optimal Control,” in Robotics and Automation (ICRA), IEEE International Conference on, 2014.

[12] S. Lengagne, J. Vaillant, E. Yoshida, and A. Kheddar, “Generation of whole-body optimal dynamic multi-contact motions,” The International Journal of Robotics Research, vol. 32, no. 9-10, pp. 1104–1119, 2013.

[13] H. Dai, A. Valenzuela, and R. Tedrake, “Whole-body Motion Planning with Simple Dynamics and Full Kinematics,” in Humanoid Robots (Humanoids), 14th IEEE-RAS International Conference on, 2014.

[14] P. M. Wensing and D. E. Orin, “Improved computation of the humanoid centroidal dynamics and application for whole-body control,” International Journal of Humanoid Robotics, vol. 13, no. 01, p. 1550039, 2016.

[15] G. Ficht and S. Behnke, “Online Balanced Motion Generation for Humanoid Robots,” in IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids), 2018.

[16] J. Arreguit, S. Faraji, and A. J. Ijspeert, “Fast multi-contact whole-body motion planning with limb dynamics,” in IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids), 2018.

[17] P. Allgeuer and S. Behnke, “Fused Angles: A representation of body orientation for balance,” in Int. Conf. on Intelligent Robots and Systems (IROS), 2015.

[18] W. Erdmann, “Geometry and inertia of the human body-review of research,” Acta of Bioengineering and biomechanics, 1999.

[19] P. Allgeuer. (2014) Matlab/Octave Rotations Library. [Online]. Available: https://github.com/AIS-Bonn/matlab_octave_rotations_lib/

[20] P. Allgeuer, H. Farazi, M. Schreiber, and S. Behnke, “Child-sized 3d printed igus humanoid open platform,” in Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on, 2015, pp. 33–40.