

Bipedal Trajectory Generation based on Combining Inertial Forces and Intrinsic Angular Momentum Rate Changes: Eulerian ZMP Resolution

Barkan Ugurlu, *Member, IEEE*, and
Atsuo Kawamura, *Fellow, IEEE*

Abstract—This paper is aimed at presenting a technique to generate feasible and dynamically-equilibrated ZMP-based CoM trajectories that can be applied to bipedal robots. In this regard, we utilize the ZMP concept in the spherical coordinate frame; so that we can fully exploit its properties as this strategy enables us to efficiently combine intrinsic angular momentum rate change terms with inertial force terms. That being said, this strategy is accompanied with certain advantages: 1) In case of bipedal walking, undesired torso angle fluctuations are more restrainable compared to other methods, in which angular momentum information is omitted or zero-referenced. 2) Composite rigid body inertia, a multi-body property of robot dynamics, can be characterized during the trajectory generation task. Thus, relatively more dynamically-consistent trajectories may be obtained. 3) The motion interference between sagittal and lateral planes is naturally included. In this paper, we mainly investigate the first two merits. Applying the method described above, we conducted bipedal walking experiments on our actual bipedal robot MARI-3. As the result, we obtained repetitive, continuous and dynamically-equilibrated walking cycles, in which undesired torso angles were well suppressed. Furthermore, ZMP errors tend to decrease since inertial parameters of the robot are characterized. In conclusion, the method is validated to be efficient in inducing less ZMP errors and in suppressing undesired torso angle variations, compared to both flywheel-superimposed and conventional ZMP-based trajectory generation methods.

Index Terms—biped robot, humanoid, bipedal walking, zmp, angular momentum.

I. INTRODUCTION

Due to inherent nonlinearities associated with humanoid robots, bipedal motion planning task is considered to be one of the most challenging -yet very intriguing- fields in the robotics literature. Generally, these robots interact with the outer environment through floating base points, i.e., through their hands and feet. This kind of unilateral contacts make them underactuated, resulting in the fact that a strong coupling occur between their dynamics and geometry [1].

In regard to this matter, humanoid robot dynamics are often abstracted by employing various modeling tools, such as inverted pendulum [2], ZMP (Zero Moment Point) [3], cart-table model [4], and so on. The main idea in this strategy is to capture the essence of dominant bipedal robot characteristics, so that researchers can simplify the multi-DoF (degrees of freedom) motion planning task into two subgroups: a) Online trajectory generation to obtain feasible motion patterns, b) Closed-loop feedback control to stabilize the robot. This approach is experimentally proved to be efficient for motion control of humanoid robots [1] [4] [5].

Nonetheless, a majority of these approaches cannot characterize the rotational inertia and associated angular momentum (intrinsic angular momentum [6]), which appear to be crucial components regarding the rotational behavior and balance of bipedal robots [7]. By the same token, conventionally used ZMP equations only consist of gravitational and inertial force terms, excluding Intrinsic Angular

Momentum Rate change¹ (IAMR) terms. Despite the fact that such an action provides a simplified model, it may not represent a certain portion of robot dynamics. For instance, our analysis on MARI-3 walking motion revealed that approximately 40% of ZMP response variations are caused by IAMR terms, while the rest is due to inertial forces. That being said, several researchers reported hands-on experiences, in which omitting IAMR terms caused following problems on robots with significant inertial characteristics (size, weight, mass distribution).

- Actual stepping motion could be unusual. The body and legs may perform twists to cancel a certain portion of angular momentum about the CoM (Center of Mass) [5] [8].
- In a similar manner, it may create undesired torso angle fluctuations [5], [8], [9], [10].
- ZMP loses rich dynamics information concerning inertial parameters. In other words, ZMP equations without IAMR terms include no rotational inertia elements [10].

In order to deal with these drawbacks, we need to pay special attention to angular momentum induced at the CoM. To this end, Kajita and his colleagues proposed resolved momentum control, in which joint rates are related with the total momenta [9]. Though its name evokes a control method, it functions to generate joint motions based on translational and angular momenta inputs. In a similar fashion, Orin and Goswami derived the centroidal momentum matrix of a general humanoid [11]. As another example, Sugihara and Nakamura enhanced boundary condition relaxation method for 3-D hopping motion planning [12]. In their ZMP-based method, IAMR terms are firstly omitted, but angular momentum constraint is included in the joint motion computation stage. The common ground in aforementioned methods is to assess the angular momentum information during joint motion generation. In contrast to this approach, we believe that angular momentum information could be processed during the CoM trajectory generation stage. Consequently, we may enrich *abstracted humanoid dynamics*-based bipedal motion planning methods and generate more feasible CoM trajectories without ignoring intrinsic angular momentum information. A slightly parallel trend regarding this goal can also be examined in [7].

Due to the facts defended above, we intend to process angular momentum information during CoM trajectory generation. Thus, IAMR terms must be combined with gravitational and inertial force terms in ZMP equations. For this purpose, a widely used improper approach is to sum up all individual links' centroidal torques [13], [14]. This kind of addition may hold valid for inertial force terms; however, individual links' centroidal torques are not additive unless they are projected to a common frame [11] [15]. In conjunction with this issue, a proper approach can be examined in [8] and [16].

Flywheel-based methods also enable researchers to approximate angular momentum computations for bipedal motion planning [17]. Their efficacy is demonstrated within the support of dynamics error compensators [18] that eliminate modeling inaccuracies. In this connection, flywheel-based methods usually represent a single rotation of the body that is superimposed within CoM trajectory scheme. Therefore, this approach may enhance the system performance for reasonably small angles. While useful in their own right, flywheel-based methods may not fully represent concurrent rotational motions, which simultaneously occur in lateral and sagittal planes during 3-D motion of a biped robot. Moreover, such approaches treat the rotational inertia as a single constant element, and it is often characterized through trial and error experimentation. Contrary to this matter, the actual rotational inertia of a 3-D bipedal robot is

¹IAMR refers the time derivative of (rotational inertia tensor) $\cdot$ (angular velocity vector) multiplication [6]; $\frac{d}{dt} \{ \tilde{I} \cdot \tilde{\omega} \}$.

B. Ugurlu is with the Department of Advanced Science and Technology, Toyota Technological Institute, 468-8511 Nagoya, Japan. (e-mail: barkanu@toyota-ti.ac.jp) A. Kawamura is with the Graduate School of Engineering, Yokohama National University, 240-8501 Yokohama, Japan. (e-mail: kawamura@ynu.ac.jp) The first author gratefully acknowledges Isil Cakir Ugurlu's helpful assistance. Authors would also like to thank Yasutaka Fujimoto, Thomas Buschmann, Shuuji Kajita, Nikos G. Tsagarakis, and Tatsuo Narikiyo for their valuable suggestions.

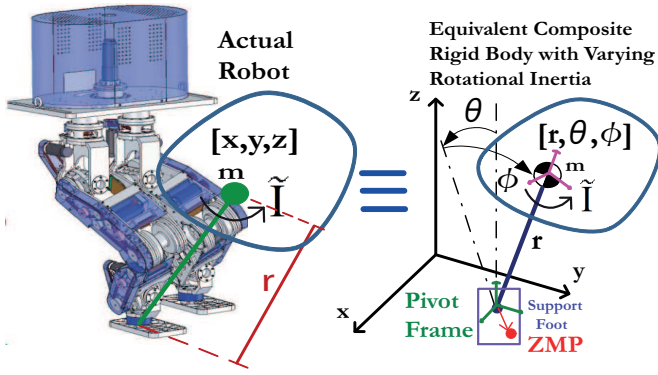


Fig. 1. Composite rigid body model and spherical coordinates on our bipedal robot MARI-3 [20]. The actual robot is considered as a 3-D composite rigid body with variant rotational inertia. Its CoM is assumed to be oriented in an identical manner to the composite body. In order to mimic the physical robot, we compute the actual robot's composite rigid body inertia matrix [19] and utilized in this model. Inertia computation algorithm processes the information of all individual links and it is updated in each control cycle. The origin of Cartesian frame and pivot frame are in fact coincided. The offset is left for illustrative purposes.

defined with tensors and its elements are functions of joint motions [19]. Taking into account of these facts, our aim is to treat rotational motion in a hypothetically more consistent manner by 2 additional actions: i) Utilizing the rotational inertia tensor of the actual 3-D robot, ii) considering the simultaneous rotational actions that occur in sagittal and lateral planes (see Fig. 1).

Considering the above discussed factors, this paper has two primary goals. The first one is to constitute an abstracted humanoid robot model, in which the rotational inertia can be characterized. Usually, point-mass models are used to simplify humanoid dynamics by considering the whole system mass is condensed at one point with no rotational inertia [2]. This issue is partially remedied in flywheel models by treating rotational inertia as a constant element for planar motion [17]. In our modeling approach, the whole robot is considered to be an equivalent 3-D composite rigid body with varying rotational inertia. In order to mimic the physical robot, we compute the actual robot's composite rigid body inertia [19] and utilize in this model.

The second goal is to compose a ZMP-based trajectory generation technique using the proposed model, for the purpose of acquiring dynamically-consistent CoM patterns. In our proposed method, ZMP equations are utilized in spherical coordinates, so that IAMR terms can be efficiently combined with inertial force terms by using Euler's equations of motion. This allows us to fully exploit ZMP equations for the trajectory generation task. What is more, it enables us to demonstrate simultaneous body rotations both in sagittal and lateral planes, as depicted in Fig. 1. Since IAMR terms are included using Euler's equations, we name it Eulerian ZMP Resolution (EZR). In this context, we would like to highlight the fact that this paper mainly focuses on the trajectory generation problem. As for real-time stabilization, we utilized a ZMP controller that is based on our previous work [21]. A competent real-time stabilizer which can cope with contact forces and unpredictable disturbances will be addressed in our next publication.

An early version of this method containing preliminary experimental results is presented in [22]. Subsequently, the method is extended by considering the time derivatives of rotational inertia elements [10]. The current paper is a significantly expanded and improved version to provide a complementary archival report. Enhancements include following; i) detailed review of the related works, ii) restructured mathematical formulation, iii) exhaustive discussions on IAMR terms,

iv) latest experimental results with thorough comparative analysis, v) further discussions on limitations of the proposed method.

The article is organized as follows. In section II, an overview on Eulerian ZMP Resolution is addressed. IAMR terms are formulated and discussed in section III. EZR-based bipedal trajectory generation is introduced in section IV. Bipedal walking experiment results are thoroughly discussed in section V. Finally, the article is concluded in section VI by stating the remarks and addressing the future directions.

II. OVERVIEW ON EULERIAN ZMP RESOLUTION

In a basic sense, EZR is a method to generate feasible ZMP-based CoM trajectories for bipedal motions that includes single support phases. Naturally, single support phases are common in both walking, jumping and running motions. As for double support phases during walking, we define proper boundary conditions to connect these phases seamlessly. Since EZR is a ZMP-based method, let us analyze ZMP equations [3] [5]:

$$X_{zmp} = \frac{\overbrace{mx(\ddot{z} + g) - m\ddot{z}x}^{\text{Inertial Forces}}}{m(\ddot{z} + g)} - \frac{\overbrace{\dot{L}_y}^{\text{IAMR}}}{m(\ddot{z} + g)} \quad (1)$$

$$Y_{zmp} = \frac{my(\ddot{z} + g) - m\ddot{z}y}{m(\ddot{z} + g)} + \frac{\dot{L}_x}{m(\ddot{z} + g)} \quad (2)$$

In (1) and (2), x , y and z stand for CoM position in the Cartesian frame, while L_x and L_y symbolize roll axis and pitch axis intrinsic angular momenta about CoM. One dot and two dots represent first and second derivatives with respect to time. Furthermore, m is the total mass and g is the gravitational acceleration. In EZR method, we are going to group ZMP equations into two parts: Inertial force terms and IAMR terms which are the first and second terms in (1) and (2), respectively. As previously stated, combining IAMR terms with inertial forces is quite challenging in Cartesian coordinates. As the result, these terms are always omitted in most of the ZMP-based approaches. In EZR, we fulfill this task using spherical coordinates while considering following two assumptions.

Assumption 1: Throughout single support phases, the robot is considered as a three dimensional composite rigid body, rotating about a fixed point (pivot frame's origin) and it is in contact with the floor through a rectangular foot as illustrated in Fig. 1. In this model, we assume that the frame which is attached to the CoM is oriented in an identical manner to the equivalent composite rigid body. In utilizing such a model, CoM configuration can be defined in the spherical coordinate frame.

The main reason for utilizing spherical coordinates is to express the equivalent composite rigid body's CoM angular velocity in compact forms, using θ and ϕ parameters within the first assumption mentioned above. In doing so, it becomes quite convenient and computationally efficient to obtain IAMR terms which are always omitted in conventional ZMP-based methods. In section III, we present a simulation study to check the consistency of our assumption by comparing it with the multi-body approach [8] [16].

For our convenience, the spherical coordinate frame is defined as follows:

- 1) Translate the frame origin r units, along $+z$ -axis.
- 2) Rotate the frame through θ , about $+y$ -axis (sagittal, pitch).
- 3) Rotate the frame through ϕ , about $+x$ -axis (lateral, roll).

Assumption 2: In addition to θ and ϕ angles, we need one more rotational action for yawing, since 3 successive rotations are needed to define the 3-D space. On the other hand, we concentrate on bipedal motion planning without turning (yawing), so that presented

definition fits well to our case. This limitation will be further addressed in section V.

Employing the aforementioned spherical coordinates, we are able to efficiently combine IAMR terms with inertial force terms that appear in (1) and (2). Henceforward, we are going to express both IAMR terms and inertial forces by using θ and ϕ parameters.

III. FORMULATING IAMR TERMS IN EZR

As previously stated, we may consider the single support phase as two successive rotations of a composite rigid body about a fixed point (pivot frame's origin). This situation is previously illustrated in Fig. 1. To this end, Euler's equations might provide us insights.

$$\dot{L}_x = I_{xx}\dot{\omega}_x + I_{xy}(\dot{\omega}_y - \omega_z\omega_x) + I_{xz}(\dot{\omega}_z + \omega_x\omega_y) + \dot{I}_{xx}\omega_x + (I_{zz} - I_{yy})\omega_y\omega_z + I_{yz}(\omega_y^2 - \omega_z^2) + \dot{I}_{xy}\omega_y + \dot{I}_{xz}\omega_z \quad (3)$$

$$\dot{L}_y = I_{xy}(\dot{\omega}_x + \omega_y\omega_z) + I_{yy}\dot{\omega}_y + I_{yz}(\dot{\omega}_z - \omega_x\omega_y) + \dot{I}_{yy}\omega_y + (I_{xx} - I_{zz})\omega_z\omega_x + I_{xz}(\omega_z^2 - \omega_x^2) + \dot{I}_{xy}\omega_x + \dot{I}_{yz}\omega_z \quad (4)$$

$$\dot{L}_z = I_{xz}(\dot{\omega}_x - \omega_y\omega_z) + I_{yz}(\dot{\omega}_y + \omega_x\omega_z) + I_{zz}\dot{\omega}_z + \dot{I}_{zz}\omega_z + (I_{yy} - I_{xx})\omega_x\omega_y + I_{xy}(\omega_x^2 - \omega_y^2) + \dot{I}_{xz}\omega_x + \dot{I}_{yz}\omega_y \quad (5)$$

In (3), (4) and (5), I_{xx} , I_{yy} and I_{zz} are moments of inertia and I_{xy} , I_{xz} and I_{yz} are products of inertia at composite CoM; $\vec{\omega} = [\omega_x \ \omega_y \ \omega_z]^T$ and $\vec{L} = [\dot{L}_x \ \dot{L}_y \ \dot{L}_z]^T$ are angular velocity and IAMR vectors, respectively. Usually, Euler's equations are defined for rigid bodies with constant rotational inertia elements. In the appendix, we presented a modification to this equation set in a way that it includes rotational inertia elements' time derivatives, as these terms are not constant in bipedal robots.

Assessing (3), (4) and (5), we may see that it is necessary to express angular velocity, $\vec{\omega}$, and angular acceleration, $\vec{\dot{\omega}}$. With the Assumption 1 in mind, it is possible to express the equivalent composite rigid body's (see Fig. 1) CoM orientation by considering 2^{nd} and 3^{rd} rotational actions while defining the spherical coordinate frame.

$$Q_s = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ \sin \theta \sin \phi & \cos \phi & -\cos \theta \sin \phi \\ -\sin \theta \cos \phi & \sin \phi & \cos \theta \cos \phi \end{bmatrix} \quad (6)$$

Utilizing a tensorial approach [15], angular velocity can be expressed in skew-symmetric form ($\omega^\dagger$).

$$\omega^\dagger = \dot{Q}_s Q_s^T = \begin{bmatrix} 0 & -\dot{\theta} \sin \phi & \dot{\theta} \cos \phi \\ \dot{\theta} \sin \phi & 0 & -\dot{\phi} \\ -\dot{\theta} \cos \phi & \dot{\phi} & 0 \end{bmatrix} \quad (7)$$

Angular velocity vector can simply be extracted from (7) as expressed in (8). Differentiating this vector, we can obtain angular acceleration vector as well.

$$\begin{aligned} \omega_x &= \dot{\phi} & \dot{\omega}_x &= \ddot{\phi} \\ \omega_y &= \dot{\theta} \cos \phi & \dot{\omega}_y &= \ddot{\theta} \cos \phi - \dot{\theta} \dot{\phi} \sin \phi \\ \omega_z &= \dot{\theta} \sin \phi & \dot{\omega}_z &= \ddot{\theta} \sin \phi + \dot{\theta} \dot{\phi} \cos \phi \end{aligned} \quad (8)$$

Inserting (8) into (3) and (4), we can obtain $\dot{L}_x$ and $\dot{L}_y$ as below.

$$\dot{L}_x = I_a \ddot{\theta} + I_{xx} \ddot{\phi} + I_t \dot{\theta}^2 - 2I_c \dot{\theta} \dot{\phi} + M_a \dot{\theta} + \dot{I}_{xx} \dot{\phi} \quad (9)$$

$$\begin{aligned} \dot{L}_y &= I_b \ddot{\theta} + I_{xy} \ddot{\phi} + I_a \dot{\theta}^2 \sin \phi - I_{xz} \dot{\phi}^2 + I_n \dot{\theta} \dot{\phi} \sin \phi \\ &+ M_b \dot{\theta} + \dot{I}_{xy} \dot{\phi} \end{aligned} \quad (10)$$

To be able to ease our computations, we utilized some sub-expressions in (9) and (10) as listed in the following.

TABLE I
MECHANICAL DATA AND IAMR APPROX. ERRORS IN RMS

Robot Name	Total Mass	Height	$\delta \dot{L}_x$, (rms)	$\delta \dot{L}_y$, (rms)
MARI-1 ² [23]	57.48 [kg]	0.98 [m]	0.04 [Nm]	0.05 [Nm]
MARI-3 [20]	33.64 [kg]	1.22 [m]	0.08 [Nm]	0.11 [Nm]
CoMan [24]	17.77 [kg]	0.69 [m]	0.21 [Nm]	0.27 [Nm]
ScBot	8.21 [kg]	0.57 [m]	0.79 [Nm]	0.88 [Nm]

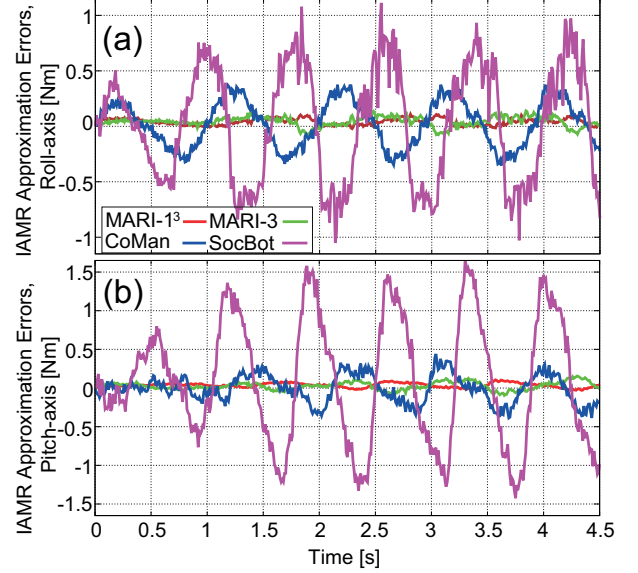


Fig. 2. Induced IAMR approximation error variations, $(\delta \dot{L}_x, \delta \dot{L}_y)$. Solid red, green, blue and magenta lines respectively indicate variations for MARI-1, MARI-3, CoMan and ScBot. Stepping frequencies are kept same for all robots. Since CoMan and ScBot have shorter legs, stride lengths are scaled accordingly.

$$\begin{aligned} I_a &= I_{xy} \cos \phi + I_{xz} \sin \phi; & M_a &= \dot{I}_{xy} \cos \phi + \dot{I}_{xz} \sin \phi; \\ I_b &= I_{yy} \cos \phi + I_{yz} \sin \phi; & M_b &= \dot{I}_{yy} \cos \phi + \dot{I}_{yz} \sin \phi; \\ I_c &= I_{xy} \sin \phi - I_{xz} \cos \phi; & I_n &= I_{xx} - I_{yy} - I_{zz}; \\ I_t &= I_{yz} \cos 2\phi - 0.5 (I_{yy} - I_{zz}) \sin 2\phi. \end{aligned} \quad (11)$$

In above calculations, rotational inertia elements and their derivatives are recursively computed in each control cycle via Balafoutis' algorithm [19], by utilizing the method presented in [10]. This algorithm processes the information of all individual bodies, in a way to construct the rotational inertia at the composite CoM. There are other algorithms which works on the same principle as well [7].

In order to check the consistency of Assumption 1, we simulated walking motion in a 3-D precise simulator [23], using the realistic models of four robots: i) MARI-1 [23]², a whole-body humanoid, b) MARI-3 [20], c) CoMan [24], a 4-years-old-child-size bipedal robot, d) ScBot, a small sized humanoid similar to that of [1]. A brief mechanical data of the robot models are given in Table I.

The main objective of this simulation is to check if (9) and (10) are capable of representing the actual IAMR computation for a bipedal robot. Having generated the walking motion, we computed induced IAMR response for roll and pitch axes, using (9) and (10). Secondly, we computed the same quantities utilizing a multi-body

²Since MARI-1 is originally made by carbon fiber reinforced material and thus quite lightweight (29 [kg] overall), we doubled each link's mass and inertia values in the simulation model, so that we obtain a model that is identical to standard humanoids, such as HRP-2 [4].

approach [8] [16]. The difference between two computations revealed IAMR approximation error variations. Results are exhibited in Fig. 2-(a) and Fig. 2-(b), respectively for roll and pitch axes IAMR approximation error variations ($\delta\dot{L}_x$, $\delta\dot{L}_y$). Table I also summarizes RMS (Root Mean Square) values of approximation errors.

As may be observed in Fig. 2 and Table I, approximation errors for MARI-1 and MARI-3 robots are truly negligible; indicating that our computations based on (9) and (10) may replace their multi-body counterparts for these robots. In the case of CoMan, approximation error seems to increase. Peak-to-peak variations is examined to be within ± 0.35 [Nm] band. Though the approximation error is still tolerable, one may debate if our computations based on (9) and (10) could provide physically consistent results for this robot. As we proceed, approximation error gets larger for ScBot. Peak-to-peak fluctuations are in the band of ± 1.5 [Nm] and ± 1.15 [Nm] for pitch and roll axes. Evidently, (9) and (10) may not replace actual multi-body IAMR calculations for this robot.

Judging by the above results, it is clear that our approach is valid for robots with relatively significant inertial properties. When looking at the robot models, MARI-1 and MARI-3 appear to be close to human size and they are not light-weight. As the robot weight and size decreases, rotational inertia becomes less significant; meanwhile approximation error gets larger. Observing this trend, we may hypothesize that the accuracy of our approach is better for robots with larger rotational inertia. Therefore, one should consider target robot inertial parameters (size, weight, mass distribution, etc.) when attempting to utilize EZR method. This limitation is further addressed in section V.

IV. EZR-BASED BIPEDAL WALKING TRAJECTORY GENERATION

A. Single Support Phase Planning

For single support phase planning, we primarily need to combine IAMR and inertial forces terms in ZMP equations. Therefore, a coordinate transformation between Cartesian frame and spherical coordinate frame is performed as follows.

$$\begin{bmatrix} x & y & z \end{bmatrix}^T = Q_s \begin{bmatrix} 0 & 0 & r \end{bmatrix}^T \quad (12)$$

$$x = r \sin \theta \quad (13)$$

$$y = -r \cos \theta \sin \phi \quad (14)$$

$$z = r \cos \theta \cos \phi \quad (15)$$

Since the ZMP-based motion generation is an underconstrained problem in the general sense, one of the parameters should be predetermined. Conventionally, researchers supply z-axis CoM trajectory (z) in advance, then solve ZMP equations for x-axis and y-axis CoM trajectories with given ZMP inputs. This predetermined z-axis CoM trajectory is usually kept constant during walking, or optimized through reinforcement learning procedures for energy efficiency [25]. In case of running or jumping, it may be defined via polynomials to meet the jumping conditions [21]. Because of these advantages, we also choose to predetermine z-axis CoM trajectory. Therefore, we perform mapping between (x, y) and (θ, ϕ) for a predetermined z parameter. That being said, r parameter can be expressed in terms of $(z, \theta$ and $\phi)$, via (15).

$$r = z \sec \theta \sec \phi \quad (16)$$

Considering (16), x , y , and their second time derivatives take the following form.

$$x = z \tan \theta \sec \phi \quad (17)$$

$$y = -z \tan \phi \quad (18)$$

$$\begin{aligned} \ddot{x} = & \tan \theta \sec \phi (\ddot{z} + (z\ddot{\theta} + 2\dot{z}\dot{\theta}) \tan \phi + z\dot{\phi}^2 (2 \sec^2 \phi - 1)) \\ & + (1 + \tan^2 \theta) \sec \phi (z\ddot{\theta} + 2\dot{\theta}(\dot{z} + z(\dot{\theta} \tan \theta + \dot{\phi} \tan \phi))) \end{aligned} \quad (19)$$

$$\ddot{y} = -(\ddot{z} \tan \phi + (1 + \tan^2 \phi)(z\ddot{\phi} + 2\dot{\phi}(\dot{z} + z\dot{\phi} \tan \phi))) \quad (20)$$

Having expressed both inertial forces and IAMR terms by using (θ, ϕ) parameters, we are going to combine them and then solve for θ and ϕ trajectories. Firstly, let us define τ_a and τ_b , which come out from cross multiplications in (1) and (2) to eliminate fractional expressions.

$$\tau_a = \overbrace{mX_{zmp}(\ddot{z} + g)}^{\text{Left Hand Sides}} = \overbrace{mx(\ddot{z} + g) - mz\ddot{x} - \dot{L}_y}^{\text{Right Hand Sides}} \quad (21)$$

$$\tau_b = mY_{zmp}(\ddot{z} + g) = my(\ddot{z} + g) - mz\ddot{y} + \dot{L}_x \quad (22)$$

Subsequently, if we insert (9), (10), (17), (18), (19) and (20) into right hand sides (RHS) of (21) and (22), we obtain the following equations.

$$\tau_a = \eta_a + \kappa_a + \gamma_a - \ddot{\theta}(I_b + J_a c_\phi) - \ddot{\phi}(I_{xy} + J_b) \quad (23)$$

$$\tau_b = \ddot{\phi}(J_a c_\theta^2 + I_{xx}) + \ddot{\theta}I_a + \eta_b + \kappa_b + \gamma_b \quad (24)$$

In (23) and (24), s_θ , $s_{2\theta}$, s_ϕ , c_θ , c_ϕ , t_θ , t_ϕ , d_ϕ , j_θ , and j_ϕ symbolize $\sin \theta$, $\sin 2\theta$, $\sin \phi$, $\cos \theta$, $\cos \phi$, $\tan \theta$, $\tan \phi$, $\cot \phi$, $\sec \theta$ and $\sec \phi$, respectively. Moreover, quadratic terms (η_a , η_b), linear terms (κ_a , κ_b) and gravity terms (γ_a , γ_b) are also grouped as below.

$$\eta_a = \dot{\theta}\dot{\phi}s_\phi(2J_a s_\theta^2 - I_n) + \dot{\phi}^2(I_{xz} - J_b d_\phi) - \dot{\theta}^2 I_a s_\phi \quad (25)$$

$$\eta_b = \dot{\theta}^2 I_t - \dot{\theta}\dot{\phi}(2I_c + J_a s_{2\theta}) \quad (26)$$

$$\kappa_a = -\dot{\theta}(M_b + M_e c_\phi) - \dot{\phi}(I_{xy} + M_d) \quad (27)$$

$$\kappa_b = \dot{\theta}M_a + \dot{\phi}(I_{xx} + M_e c_\theta^2) \quad (28)$$

$$\gamma_a = mgz t_\theta j_\phi \quad (29)$$

$$\gamma_b = -mgz t_\phi \quad (30)$$

Following subexpressions are frequently encountered in equations (23)-(30). Therefore, grouping them in the following manner increases computational efficiency as done in (11).

$$J_a = m(zj_\theta j_\phi)^2 \quad M_e = 2mz(j_\theta j_\phi)^2(\dot{z} + z(\dot{\theta} t_\theta + \dot{\phi} t_\phi)) \quad (31)$$

$$J_b = 0.5J_a s_{2\theta} s_\phi \quad M_d = 0.5M_e s_{2\theta} s_\phi \quad (32)$$

Having designated feasible ZMP referential inputs (X_{zmp} , Y_{zmp}), and using predetermined z ; left hand sides (LHS) of (21) and (22) can easily be calculated. Hence, we can obtain τ_a and τ_b . As next step, we arrange (23) and (24) in a way to obtain solution for θ and ϕ parameters.

$$\ddot{\theta} = \frac{(I_{xy} + J_b)(\eta_b + \kappa_b - \tau_b + \gamma_b) - (J_a c_\theta^2 + I_{xx})(\tau_a - \eta_a - \kappa_a - \gamma_a)}{(J_a c_\phi + I_b)(J_a c_\theta^2 + I_{xx}) - I_a(I_{xy} + J_b)} \quad (33)$$

$$\ddot{\phi} = \frac{I_a(\tau_a - \eta_a - \kappa_a - \gamma_a) + (J_a c_\phi + I_b)(\tau_b - \eta_b - \kappa_b - \gamma_b)}{(J_a c_\phi + I_b)(J_a c_\theta^2 + I_{xx}) - I_a(I_{xy} + J_b)} \quad (34)$$

Equations (33) and (34) describe a pair of coupled 2^{nd} order differential equations. These equations can be solved concurrently by employing Runge-Kutta method with adaptive stepsize control [26] if following parameters are designated.

- z-axis CoM trajectory: z

- Referential ZMP inputs: X_{zmp} and Y_{zmp}
- Initial values: $\theta_0, \phi_0, \dot{\theta}_0$ and $\dot{\phi}_0$
- Single support Phase Time Interval: $(t_{start}, t_{start} + t_{ss})$

Generally speaking, a numerical differential equation solver should change its stepsize to perform adaptive control regarding its own process. In doing so, it is possible to solve a set of differential equations for a predetermined accuracy with the least possible computational effort. Adaptive stepsize Runge-Kutta algorithm can handle this task; it automatically manages the trade-off between computational cost and accuracy while solving (33) and (34). For details, refer to [26].

Having obtained θ and ϕ trajectories through the numerical solution of (33) and (34) within the above-mentioned manner, we have accomplished the CoM trajectory generation task without omitting IAMR terms. We needed these spherical coordinate frame parameters throughout the CoM trajectory generation process, for the purpose of combining IAMR terms with inertial force terms in ZMP equations. As we achieved our goal, we may perform a conversion back to the Cartesian frame using (17) and (18).

In real-time applications, $(\theta_0, \phi_0, \dot{\theta}_0, \dot{\phi}_0)$ are set uniquely with a given set of $(z, X_{zmp}, Y_{zmp}, t_{ss})$ for single support phases. In doing so, we obtain symmetrical CoM patterns which correspond following conditions [24] to provide convenience in trajectory planning.

- In x-axis, initial and terminal CoM velocities are equal to each other. Hence, the robot firstly decelerates, and then reaches to zero velocity momentarily in the middle and starts accelerating in a sequential manner.
- In y-axis, initial and terminal CoM positions are equal to each other. Moreover, terminal velocity equals to initial velocity with a minus sign. For instance, let us consider the beginning of a single support phase. The CoM firstly starts swaying from a nonzero position with a nonzero initial velocity. It starts decelerating and momentarily reaches to zero velocity in the middle. Subsequently, it starts accelerating in the opposite direction and swings back to initial position with the same velocity with a minus sign.

B. Double Support Phase Planning

Double support phases are planned by using polynomials, in a way to smoothly connect subsequent single support phases both in position, velocity and acceleration levels. As stated in the previous subsection, we set single support initial values $(\theta_0, \phi_0, \dot{\theta}_0, \dot{\phi}_0)$ in a way to obtain symmetrical CoM patterns during single support phases. Boundary conditions of a double support phase are determined by considering these symmetry properties and mechanical capabilities of the robot. Since double phase periods are relatively short, e.g. only 16% of the whole period, we followed this strategy as it worked without any problem in practice.

Nevertheless, one may utilize EZR during double support phases as well. In this scenario, the pivot frame may be substituted as the middle point of foot sole center positions. Afterwards, identical procedures can be followed to plan double support phase motion.

C. Overall Motion Generation and Control Scheme

Fig. 3 summarizes the overall motion generation and control algorithms. In this figure, subscripts *ref* and *res* indicates referential and response values. Firstly, we designate feasible ZMP inputs by considering the support polygon of MARI-3. Afterwards, referential ZMP input is compared with the actual ZMP measurement. Due to the ZMP error, reference values are updated in the ZMP controller block which is designated in a similar manner as performed in [21]. Moreover, ZMP is measured by 6-axis F/T (Force/Torque) sensors,

using the method proposed in [27]. Subsequently, the updated ZMP values and predetermined z-axis CoM trajectory are inserted to (33) and (34) to obtain a solution for θ and ϕ trajectories, in accordance with the proposed trajectory generation method. As next step, we perform a conversion back to Cartesian frame; namely from (θ, ϕ) to (x, y) , by using (17) and (18). Together with the predetermined z-axis trajectory, we finally obtain all the necessary components of CoM trajectory by using Eulerian ZMP Resolution, in which IAMR terms are characterized. Following that, CoM and swing leg trajectories are plugged into inverse kinematics block [22], together with the orientation constraint. In this inverse kinematics calculation, CoM and swing leg trajectories are processed simultaneously, so that the CoM motion is not influenced by swing leg motion. It is also functional during double support phases, in which one of the support feet is considered as the pseudo swing foot with stationary motion. As the output of inverse kinematics, we obtain all referential joint angle values. Referential joint angles are individually treated in local servo blocks, where position control, velocity control and disturbance observers are applied [20] [28] to obtain command torque, τ_{cmd} .

Note that generated joint command parameters are subject to following actuator limitations, where superscripts $+$ and $-$ denote upper and lower boundaries.

$$\tau^- \leq \tau_{cmd} \leq \tau^+ ; \quad q^- \leq q_{ref} \leq q^+ ; \quad \dot{q}^- \leq \dot{q}_{ref} \leq \dot{q}^+ \quad (35)$$

In practice, motion generation parameters are empirically assigned in a way not to violate actuator limitations. Furthermore, the frictional force should always exist within the friction cone [23].

$$\sqrt{F_{rx}^2 + F_{ry}^2} \leq \mu_t F_{rz} ; \quad \sqrt{F_{lx}^2 + F_{ly}^2} \leq \mu_t F_{lz} \quad (36)$$

$$|\eta_{rz}| \leq \mu_r F_{rz} ; \quad |\eta_{lz}| \leq \mu_r F_{lz} \quad (37)$$

In (36) and (37), F_{rj} and F_{lj} ($j = x, y, z$) denote ground reaction force for right and left feet. η_{rz} and η_{lz} are induced yaw moments; μ_t and μ_r are friction constants. In MARI-3, we placed custom made rubber foot pads under the feet soles, so that μ_t and μ_r are sufficiently large to meet above conditions. Additionally, undesired yaw moment regulation is applied [29] to help reducing η_{rz} and η_{lz} .

D. Stability Discussions

In order to fully exploit ZMP equations by considering IAMR terms, we come up with eqs. (33) and (34), for the purpose of dynamically-consistent bipedal trajectory generation. This approach, on the other hand, inevitably makes these motion equations highly nonlinear without any analytical solution. Moreover, they include rotational inertia elements which are to be updated recursively in each control cycle. Due to these facts, it may not be possible to perform nonlinear stability analysis regarding the ZMP controller block ($C(s)$). Therefore, we implemented a system identification routine on the actual biped robot to have an approximate model of the plant $G(s)$, which is defined between ZMP referential input and ZMP response (see Fig. 3). We followed a similar strategy that is proposed in [30] and [31]. As the result, an approximate plant model, which fits with the actual model for about 84% is obtained. As next step, controller gain tuning and stability analysis are carried out considering the estimated model.

Another important issue in bipedal motion stability is the convergence towards periodic gaits. On this matter, Chevallereau and her colleagues proposed a control strategy in which the position of ZMP and joint angles are simultaneously regulated to guarantee periodicity [32]. Their control strategy presented in [32] somewhat

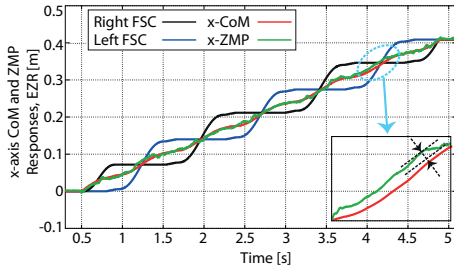


Fig. 4. x-axis CoM and ZMP Responses, EZR

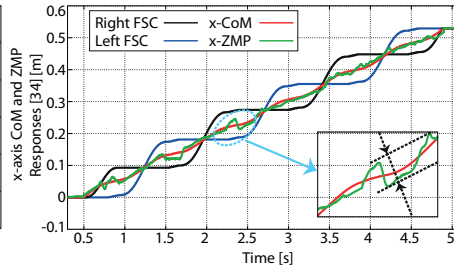


Fig. 5. x-axis CoM and ZMP Responses, [34]

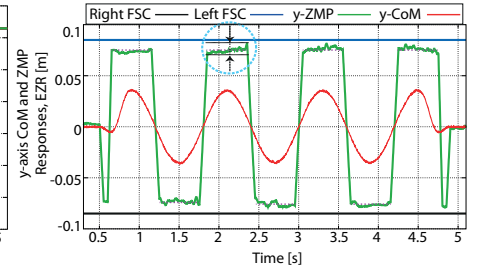


Fig. 6. y-axis CoM and ZMP Responses, EZR

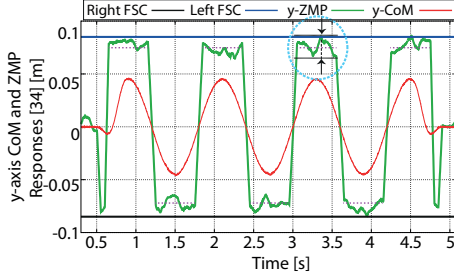


Fig. 7. y-axis CoM and ZMP Responses, [34]

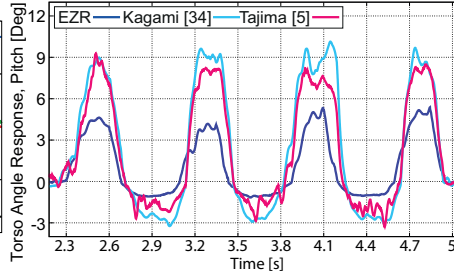


Fig. 8. Pitch axis Torso Angle

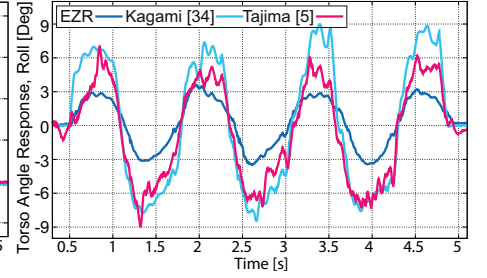


Fig. 9. Roll axis Torso Angle

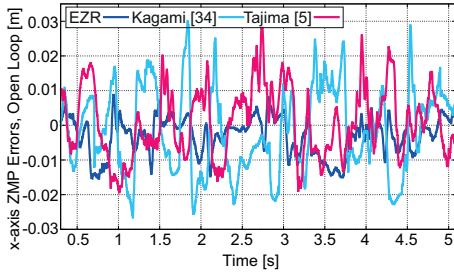
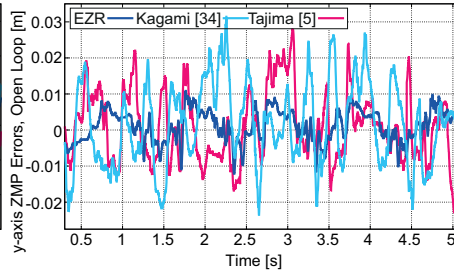
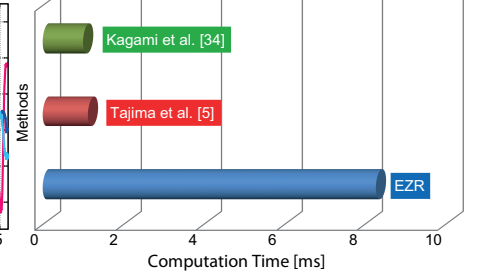
Fig. 10. Open-loop X_{zmp} ErrorsFig. 11. Open-loop Y_{zmp} Errors

Fig. 12. Average Computation Time

for roll axis was recorded between ± 9 degrees. If we make use of Tajima's trajectory generator, pitch axis torso angle fluctuates between $-3 \sim +9$ degrees and roll axis torso angle alters between ± 6 degrees. In comparison, torso angle variations decreased to $-1 \sim +5$ degrees for pitch axis, and ± 3 degrees for roll axis, when employing EZR. As the result, peak-to-peak torso angle variations decreased for about 58% in pitch axis and 67% in roll axis, comparing to Kagami et al.'s method. If we compare EZR performance with respect to Tajima et al.'s method, we obtain 50% decrease, both in pitch and roll axes torso angle variations. Considering these results, we may claim that EZR is efficient in suppressing such undesired torso angle variations, as it is able to characterize the IAMR terms within a relatively more dynamically-consistent manner.

Additionally, it may be argued that the ZMP fluctuation phenomenon also depends on the ZMP control strategy. Even though we used the same type of controller and tune the gains for the best performance, the comparison of ZMP fluctuations can still be debated. Therefore, we conducted additional walking experiments in which ZMP feedback and ZMP controller ($C(s)$) are cancelled in both cases. The robot performed bipedal walking using the same parameters as above; within a *ZMP-wise open-loop* control strategy, where servo controllers continued operation. In order to compare the performances of both methods, we recorded x-axis and y-axis ZMP errors as shown in Fig. 10 and Fig. 11. In these figures, solid blue, cyan, and red lines symbolize ZMP error values, respectively for EZR, Kagami et al.'s and Tajima et al.'s trajectory generators.

Comparing EZR performance with respect to Kagami's method, we obtained a 55% decrease in x-axis and 53% decrease in y-axis for RMS values. When it comes to Tajima et al.'s method, ZMP errors reduced for about 36% in x-axis and 32% in y-axis for RMS values. Based on these results, we may claim that EZR is more advantageous as all the conditions are kept exactly identical in benchmarking experiments. Since EZR resolves ZMP equations without ignoring the rotational inertia and therefore IAMR terms, modeling errors are anticipated to be comparatively smaller; resulting in generating more dynamically-consistent trajectories.

Fig. 12 describes the average computation time per cycle for both methods. The computation system is a Pentium III with a 1.6 GHz processor and 2 GB memory. The average computation time values are calculated to be 8.28 [ms], 1.00 [ms] and 1.02 [ms], respectively for EZR, Kagami's and Tajima's methods. Therefore, we may claim that the computation speed is 8.3 times faster while utilizing Kagami's method. Tajima's method also handles motion generation task 8.2 times faster. This result comes as no surprise, because EZR includes additional algorithms such as real-time computation of composite rigid body inertia. While utilizing EZR, one may decrease modeling errors; however, the resulting motion equations become more complex. This outcome has a price, being paid using the computation time. On the other hand, rapid growth of computer technology may enable us to fasten the computation of EZR faster. We further discuss this limitation in the next subsection.

B. Limitations

As discussed previously, one of the major drawbacks of EZR lies within the heavy computation burden. Even though contemporary advanced computation systems may remedy this problem up to a certain extent, the authors believe that one should utilize EZR when the inertial properties (size, weight, mass distribution, etc.) of the target robot are significant. Nevertheless, our analysis in section III also revealed that EZR is prone to modeling errors for small-sized robots. Keeping this in mind, a relatively small-size humanoid robot can perfectly be manipulated using off-the-shelf conventional techniques whereas the utilization of EZR is unnecessarily high-priced. On the contrary, for robots with heavy legs and significant inertial properties, EZR may enhance the system performance with a cost of additional computation time. The choice should be made by considering the target robot and the application objectives.

In [4], Kajita and his colleagues proposed a preview control method which appears to integrate trajectory generation method with feedback control, through the utilization of cart-table model. In this model, the robot is considered as a point mass with no rotational inertia, and the vertical CoM position is kept constant. By considering this *special case*, ZMP equations have only constant parameters (z , g), and thus can be expressed in state-space form; allowing them to utilize preview control. On the other hand, this task cannot be realized in EZR as it includes IAMR terms, which consists of rotational inertia elements. Being composite rigid bodies, these inertia elements are not constant (i.e., joint state-dependent), and they are recursively computed in every cycle as discussed previously. Therefore, preview control action for EZR motion equations does not seem to be applicable. The very same matter also prevents us to perform a thorough stability analysis (without a system identification procedure), in which the role of ZMP controller could be clarified. Possible contributions on these matters are welcome.

Additionally, this paper does not include the EZR formulation when the robot is walking in a circle, namely the yawing (cornering) motion. Usually, this cornering angle is provided by a high-level motion planner -such as [35]- and therefore given beforehand. Let us call this *yaw* angle as ψ . One can easily include ψ angle as the 3rd rotation in (6). Following the same procedures, EZR motion equations can be re-expressed with this additional ψ angle and its time derivatives ($\dot{\psi}$, $\ddot{\psi}$). Afterwards, trajectory generation task can be executed in the same way since ψ parameter is provided beforehand, just like the vertical CoM trajectory (z) parameter. Hence, this issue may not appear to be an actual drawback in EZR method.

VI. CONCLUSIONS AND FUTURE WORK

To sum up, we propose a systematic way of generating ZMP-based bipedal motion trajectories that is on the basis of combining inertial forces and intrinsic angular momentum rate changes. As the proposed technique is able to characterize the rotational inertia and intrinsic angular momentum rate changes, it potentially enriches the bipedal motion generation methods which are based on abstracted humanoid dynamics.

We validated the proposed method by performing bipedal walking experiments. In these experiments, we obtained repetitive and dynamically-equilibrated walking cycles in which induced ZMP fluctuations and ZMP errors are reduced, compared to similar methods in which IAMR terms are either not characterized [34] or approximated via a flywheel model [5]. Moreover, pitch and roll axes' torso angle variations are dramatically decreased as well. Based on these comparative analyses, we conclude that EZR potentially performs superior performance over conventional ZMP-based methods in suppressing undesired ZMP errors and torso angles variations. This property of

EZR definitely enables us to obtain more feasible and dynamically-consistent bipedal motion cycles.

An important aspect in generating stable and dynamically-consistent motion lies within the proper handling of contact forces. In doing so, overall performance can be enhanced during phase transitions and it may increase robustness against unpredictable disturbance forces. In response to that matter, we are currently developing an active compliance control scheme, in which force constraints are also imposed to the system in a way to handle contact forces and external disturbances. Furthermore, EZR is able to represent the motion interference between sagittal and lateral planes through IAMR terms. Thus, we believe that it is also worthy to exploit this motion interference from the control implication point of view.

APPENDIX

EULER'S EQUATION OF MOTION - REVISITED

Referring [6] and [36], we may differentiate intrinsic angular momentum as in the following.

$$\vec{L} = \tilde{I} \vec{\omega} \quad (38)$$

$$\dot{\vec{L}} = \tilde{I} \dot{\vec{\omega}} + \tilde{\dot{I}} \vec{\omega} \quad (39)$$

To be able to differentiate $\tilde{I}$ tensor, we employ dyadic notation.

$$\tilde{I} = \sum_{i=1}^3 \sum_{j=1}^3 I_{ij} \vec{e}_i \vec{e}_j \quad (40)$$

$$\tilde{\dot{I}} = \sum_{i=1}^3 \sum_{j=1}^3 \left[\dot{I}_{ij} \vec{e}_i \vec{e}_j + I_{ij} \dot{\vec{e}}_i \vec{e}_j + I_{ij} \vec{e}_i \dot{\vec{e}}_j \right] \quad (41)$$

In (40), $\vec{e}_i$ and $\vec{e}_j$ are unit vectors. For rigid bodies, rate change of rotational tensor elements, $\dot{I}_{ij}$, are equal to zero. Since our interest is composite rigid bodies, inertia elements' derivatives may not be assigned to zero, $\dot{I}_{ij} \neq 0$. Diverging from the solution explained in [36], we define $\dot{I}^*$, element-wise time derivative of inertia matrix.

$$\dot{I}^* = \begin{bmatrix} \dot{I}_{xx} & \dot{I}_{xy} & \dot{I}_{xz} \\ \dot{I}_{xy} & \dot{I}_{yy} & \dot{I}_{yz} \\ \dot{I}_{xz} & \dot{I}_{yz} & \dot{I}_{zz} \end{bmatrix} \quad (42)$$

Furthermore, unit vectors are also rotating with the equivalent composite rigid body, which is assumed to be oriented in an identical manner with CoM. In this case, following equalities hold valid [36].

$$\dot{\vec{e}}_i = \vec{\omega} \times \vec{e}_i; \quad \dot{\vec{e}}_j = -\vec{e}_j \times \vec{\omega} \quad (43)$$

Combining (42) and, (43) in (41), $\tilde{\dot{I}}$ is obtained.

$$\tilde{\dot{I}} = \dot{I}^* + \vec{\omega} \times \tilde{I} - \tilde{I} \times \vec{\omega} \quad (44)$$

Inserting (44) into (39),

$$\dot{\vec{L}} = \tilde{I} \cdot \dot{\vec{\omega}} + \left(\dot{I}^* + \vec{\omega} \times \tilde{I} - \tilde{I} \times \vec{\omega} \right) \cdot \vec{\omega}, \quad (45)$$

and considering $\left(\tilde{I} \times \vec{\omega} \right) \cdot \vec{\omega} = 0$, final derivation can be acquired.

$$\dot{\vec{L}} = \tilde{I} \cdot \dot{\vec{\omega}} + \left(\dot{I}^* + \vec{\omega} \times \tilde{I} \right) \cdot \vec{\omega} \quad (46)$$

If we open up (46) row by row, eqs. (3), (4) and (5) can be obtained. This set of equations appears to be a generalized version of Euler's equations of motion for composite rigid bodies, having rotational inertia elements' first time derivatives.

REFERENCES

- [1] T. Sugihara, and Y. Nakamura, "Boundary condition relaxation method for stepwise pedipulation planning of biped robots", in *IEEE Transactions on Robotics*, vol. 25, no. 3, 2009, pp. 658-669.
- [2] 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 *Proc. IEEE Conf. Intelligent Robots and Systems*, Maui, US, 2001, pp. 239-246.
- [3] M. Vukobratovic, and B. Borovac, "- Zero Moment Point - Thirty five years of its life", in *Int. Journal of Humanoid Robotics*, Vol 1, No. 1, 2004, pp. 157-173.
- [4] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control," in *Proc. IEEE Conf. Robotics and Automation*, Taipei, Taiwan, 2003, pp. 1620-1626.
- [5] R. Tajima, D. Honda, and K. Suga, "Fast running experiments involving a humanoid robot", in *Proc. IEEE Conf. Robotics and Automation*, Kobe, Japan, 2009, pp. 1571-1576.
- [6] R. Featherstone, "Rigid Body Dynamics Algorithms", *Springer*, 2008.
- [7] A. Goswami, "Reaction mass pendulum (RMP) model for humanoid robot gait and balance control", in *Humanoid Robots*, (Editor: Ben Choi), In-Tech Publishing, 2009, pp. 167-186.
- [8] T. Buschmann, "Simulating and Control of Biped Walking Robots", *PhD Dissertation*, Technical University of Munich, Germany, November 2010. [Online]: <http://www.amm.mw.tum.de/1/institute/personen/wissmitarbeiter/thomas-buschmann/> [Accessed: 23 July 2012]
- [9] S. Kajita, K. Kaneko, M. Morisawa, S. Nakaoka, and H. Hirukawa, "ZMP-based biped running enhanced by toe springs," in *Proc. IEEE Conf. Robotics and Automation*, Rome, Italy, 2007, pp. 3963-3969.
- [10] B. Ugurlu, and A. Kawamura, "Bipedal walking trajectory based on ZMP and Euler's Equations of Motion," in *Proc. IEEE Conf. Humanoid Robotics*, Nashville, US, 2010, pp. 468-473.
- [11] D. E. Orin, and A. Goswami, "Centroidal momentum matrix of a humanoid robot: Structure and properties", in *Proc. IEEE Conf. Intelligent Robots and Systems*, Paris, France, 2008, pp. 653-659.
- [12] T. Sugihara, and Y. Nakamura, "Enhancement of boundary condition relaxation method for 3D hopping motion planning of biped robots", in *Proc. IEEE Conf. Intelligent Robots and Systems*, San Diego, US, 2007, pp. 444-449.
- [13] O. Kwon, and J. H. Park, "Asymmetric trajectory generation and impedance control for running of biped robots", in *Autonomous Robots*, Vol. 26, No. 1, 2009, pp. 47-78.
- [14] M. Popovic, A. Goswami, and H. Herr, "Ground reference points in legged locomotion: Definitions, biological trajectories and control implications", in *Int. Journal of Robotics Research*, vol. 24, no. 12, 2005, pp. 1013-1032.
- [15] C. A. Balafoutis, and R. V. Patel, "Dynamic Analysis of Robot Manipulators: A Cartesian Tensor Approach", in *Kluwer Academic Pub.*, 1991.
- [16] W. Suleiman, F. Kanehiro, E. Yoshida, J. P. Laumond, and A. Monin, "Time parameterization of humanoid-robot paths", in *IEEE Transactions on Robotics*, vol. 26, no. 3, 2010, pp. 458-468.
- [17] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture Point: A Step toward Humanoid Push Recovery", in *Proc. IEEE Conf. Humanoid Robotics*, Genoa, Italy, 2006, pp. 200-207.
- [18] T. Takenaka, T. Matsumoto, and T. Yoshiike, "Real time motion generation and control for biped robot -3rd report: Dynamics error compensation", in *Proc. IEEE Conf. Intelligent Robots and Systems*, St. Louis, US, 2009, pp. 1594-1600.
- [19] C. A. Balafoutis, and R. V. Patel, "Efficient computation of manipulator inertia matrices and the direct dynamics problem", in *IEEE Transactions on Systems, Man and Cybernetics*, vol. 19, no. 5, 1991, pp. 1313-1320.
- [20] C. Zhu, and A. Kawamura, "The development of biped robot MARI-3 for fast walking and running", in *Proc. IEEE Conf. Humanoid Robotics*, Genoa, Italy, 2006, pp. 599-604.
- [21] B. Ugurlu, and A. Kawamura, "ZMP-based online jumping trajectory generation for a one-legged robot," in *IEEE Transactions on Industrial Electronics*, vol.57, no.5, 2010, pp. 1701-1709.
- [22] B. Ugurlu, and A. Kawamura, "Eulerian ZMP resolution based bipedal walking: Discussions on the intrinsic angular momentum rate change about center of mass," in *Proc. IEEE Conf. Robotics and Automation*, Anchorage, US, 2010, pp. 4218-4223.
- [23] Y. Fujimoto, and A. Kawamura, "Simulation of an autonomous biped walking robot including environmental force interaction", in *IEEE Robotics and Automation Magazine*, vol. 5, no. 2, 1998, pp. 33-42.
- [24] B. Ugurlu, N. G. Tsagarakis, E. Spyarakos-Papastravridis, and D. G. Caldwell, "Compliant joint modification and real-time dynamic walking implementation on bipedal robot cCub", in *Proc. IEEE Conf. Mechatronics*, Istanbul, Turkey, 2011, pp. 833-838.
- [25] P. Kormushev, B. Ugurlu, S. Calinon, N. G. Tsagarakis and D. G. Caldwell, "Bipedal walking energy minimization by reinforcement learning with evolving policy parameterization", in *Proc. IEEE Conf. Intelligent Robots and Systems*, San Francisco, US, 2011, pp. 318-324.
- [26] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing 2nd Edition*, Cambridge University Press, 1992.
- [27] Q. Li, A. Takanishi, and I. Kato, "A biped walking robot having a ZMP measurement system using universal force-moment sensors," in *Proc. IEEE Conf. Intelligent Robots and Systems*, Tokyo, Japan, 1991, pp. 1568-1573.
- [28] K. Ohnishi, M. Shibata, and T. Murakami, "Motion control for advanced mechatronics", in *IEEE Transactions on Mechatronics*, vol. 1, no. 1, 1996, pp. 56-67.
- [29] B. Ugurlu, T. Hirabayashi, and A. Kawamura, "A unified control frame for stable walking", in *Proc. IEEE Conf. Industrial Electronics and Control*, Porto, Portugal, 2009, pp. 4167-4172.
- [30] J. Kim, I. Park, and J. Oh, "Experimental realization of dynamic walking of the biped humanoid robot KHR-2 using zero moment point feedback and inertial measurement", in *Advanced Robotics*, vol. 20, no. 6, 2006, pp. 707-736.
- [31] B-K. Cho, J-H. Kim, and J. Oh, "Online balance controllers for a hopping and running humanoid robot", in *Advanced Robotics*, vol. 25, no. 9-10, 2011, pp. 1209-1225.
- [32] C. Chevallereau, D. Djoudi, and J. W. Grizzle, "Stable bipedal walking with foot rotation through direct regulation of the zero moment point", in *IEEE Transactions on Robotics*, vol. 24, no. 2, 2008, pp. 390-401.
- [33] H. Asada, and J. Slotine, "Robot Analysis and Control", Sect. 6.2, Position control: Why local schemes work, John Wiley & Sons Inc., 1986, pp. 136-139.
- [34] S. Kagami, T. Kitagawa, K. Nishiwaki, T. Sugihara, M. Inaba, and H. Inoue, "A fast dynamically equilibrated walking trajectory generation method of humanoid robot", in *Autonomous Robots*, vol. 12, no. 1, 2002, pp. 71-82.
- [35] E. Yoshida, J. P. Laumond, C. Esteves, O. Kanoun, T. Sakaguchi, and K. Yokoi, "Whole-body locomotion, manipulation and reaching for humanoids," in *Lecture Notes in Computer Science*, vol. 5277, 2008, pp. 210-221.
- [36] D. T. Greenwood, "Advanced Dynamics", in *Cambridge University Press*, 2003.