2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)
Beijing, China, November 6-9, 2018

New Cross-step Enabled Configurations for Humanoid Robot

Songyan Xin, Chengxu Zhou, Nikos Tsagarakis

***Abstract*— This paper explores two new configurations for humanoid robot balancing and locomotion. Centroidal momentum manipulability analysis has been performed to study the features of the newly proposed configurations. Numerical simulations show that they outperform the regular ones in terms of angular momentum manipulability. More than that, the new configurations allow the humanoid robot to perform cross-step motions which is usually risky or mechanically impossible for most existing robots. However, cross-step introduces non-convex feasible region which makes it difficult to be incorporated into our existing step planner. Therefore, a simple heuristic has been proposed to help choosing a sub-convex region for the step planner. To validate the cross-step movement, walking simulations have been performed.**

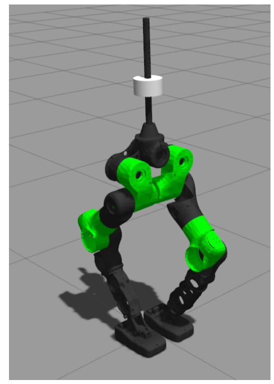
I. Introduction

For a humanoid robot, maintaining its balance is usually the first priority to guarantee. Many criteria has been proposed to evaluate its stability, and thus help designing of balance or locomotion controller. The most commonly used dynamic stability criteria is zero moment point (ZMP) [1], [2] or center of pressure (CoP), it is required to stay inside the support polygon for all time. Foot rotation indicator (FRI) [3] requires the foot has no rotation. Zero Rate of change of Angular Momentum (ZRAM) [4] guarantees rotationally stability. Capture point (CP) [5] defines a point on the ground where the robot can step to in order to bring itself to a complete stop. All these criteria summarizes the robot stability on a reduced dimension geometry point and this compression unavoidably cause the loss of information. For example, infinite configurations which are stable could ended up with the same ZMP (or CoP).

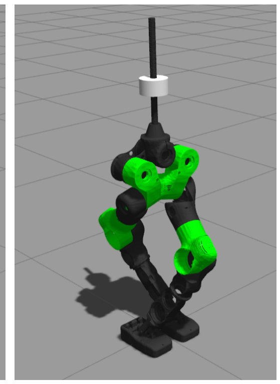
Most model-based balancing or locomotion planner use a simplified model to represent the essence of a high degree of freedom multi-rigid-body system. Based on the template model, planner often generates Center of Mass (CoM) and end-effector references for the humanoid robot to track. The ability to closely track those references becomes extremely important for system controllability and stability. Manipulability of end-effector is proposed for measuring this ability and it has been well studied [6]–[8]. Correspondingly, this manipulability concept has been extended to ZMP point [9] and CoM point [10] [11] [12] [13]. Furthermore, centroidal momentum manipulability concept [14] has been proposed to quantify system linear and angular momentum manipulability.

In this paper, two new configurations for humanoid robot have been proposed first as shown in Figure 1. Centroidal momentum manipulability analysis has been performed on these two configurations, as well as two other conventional ones. Based on the newly proposed configurations, cross-step possibility has been explored. In the end, walking simulations have been performed to show the viability of cross-step movement.

Department of Advanced Robotics, Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova, Italy.
Email: songyan.xin@iit.it



(a) *forward/backward*



(b) *twist*

Fig. 1: Two new configurations for humanoid robot which enables it to perform cross-step. The left one is called *forward/backward* configuration: left knee bending forward and right knee bending backward. The right one is called *twist* and can be achieved from the left one by one more crossing legs action.

II. Centroidal Momentum Manipulability

The Centroidal Momentum Matrix (CMM) relates the robot's generalized velocities to its centroidal momentum [14]:

$$\boldsymbol{h} = \boldsymbol{A}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{1}$$

where $\boldsymbol{h} \in \mathbb{R}^{6\times 1}$ is the centroidal momentum, $\boldsymbol{A} \in \mathbb{R}^{6\times(6+n)}$ is the CMM, $\dot{\boldsymbol{q}} \in \mathbb{R}^{(6+n)\times 1}$ is the generalized joint velocity which consists of the floating-base velocity $\dot{\boldsymbol{q}}_b = [\boldsymbol{v}_b, \boldsymbol{\omega}_b] \in \mathbb{R}^{6\times 1}$ and actuated joint velocity $\dot{\boldsymbol{q}}_a \in \mathbb{R}^{n\times 1}$.

Centroidal momentum manipulability and corresponding ellipsoid are also proposed in [14]. Due to the scale disparities between the linear and angular part of the system momentum, it is preferred to construct two ellipsoids separately. More specifically, (1) can be expanded:

$$\begin{bmatrix} \boldsymbol{l} \\ \boldsymbol{k} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_l \\ \boldsymbol{A}_k \end{bmatrix} \dot{\boldsymbol{q}} = \begin{bmatrix} \boldsymbol{A}_{lb} & \boldsymbol{A}_{la} \\ \boldsymbol{A}_{kb} & \boldsymbol{A}_{ka} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{q}}_b \\ \dot{\boldsymbol{q}}_a \end{bmatrix} \tag{2}$$

where $\boldsymbol{l} \in \mathbb{R}^{3\times 1}$, $\boldsymbol{k} \in \mathbb{R}^{3\times 1}$ are centroidal linear momentum and angular momentum, $\boldsymbol{A}_l \in \mathbb{R}^{3\times(6+n)}$ and $\boldsymbol{A}_k \in$

$\mathbb{R}^{3\times(6+n)}$ are corresponding linear and angular momentum matrix. The subscript b and a indicate the base related part and configuration related part of corresponding momentum matrix. More specifically, $\boldsymbol{A}_{lb} \in \mathbb{R}^{3\times 6}$ maps floating base velocity to system linear momentum and $\boldsymbol{A}_{la} \in \mathbb{R}^{3\times n}$ maps the actuated joint velocity part.

Given the matrix $\boldsymbol{A}_l$, the linear momentum manipulability can be calculated:

$$\omega_l = \sqrt{det(\boldsymbol{A}_l \boldsymbol{A}_l^T)} \tag{3}$$

where $det(*)$ denotes the determinant operation, the index ω_l measures the ability of transferring generalized joint velocity $\dot{\boldsymbol{q}}$ to system linear momentum $\boldsymbol{l}$. Since $\boldsymbol{A}_l$ is related to the $\boldsymbol{q}$, this index is also configuration related. This index is just a scaler indicator, more information can be visualized by constructing a ellipsoid from the matrix $\boldsymbol{A}_l$ with singular value decomposition (SVD),

$$\boldsymbol{A}_l = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{4}$$

where

$$\boldsymbol{U} = \begin{bmatrix} \boldsymbol{u}_x & \boldsymbol{u}_y & \boldsymbol{u}_z \end{bmatrix} \tag{5}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_x & 0 & 0 & ... \\ 0 & \sigma_y & 0 & ... \\ 0 & 0 & \sigma_z & ... \end{bmatrix} \tag{6}$$

$$\boldsymbol{V}^T = \begin{bmatrix} \boldsymbol{v}_1^T & \boldsymbol{v}_2^T & ... & \boldsymbol{v}_m^T \end{bmatrix}^T \tag{7}$$

The principle axes of the ellipsoid are $\sigma_x \boldsymbol{u}_x$, $\sigma_y \boldsymbol{u}_y$ and $\sigma_z \boldsymbol{u}_z$. It is worth noting that the manipulability can be also calculated from singular values $\omega_l = \sigma_x \sigma_y \sigma_z$. The same calculation also applies for the angular momentum matrix $\boldsymbol{A}_k$ and all sub-matrices in (2).

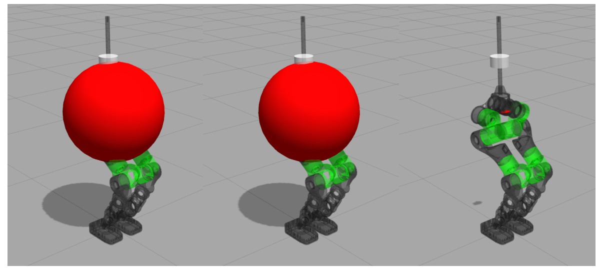
A. Manipulability Contribution

As mentioned before, the system momentum is contributed from floating base velocity $\dot{\boldsymbol{q}}_b$ and actuated joint velocity $\dot{\boldsymbol{q}}_a$. The previous part can be interpreted as base movement related contribution to system momentum, and the later part can be treated as body movement related contribution. In general, they contribute differently to system momentum. We are going to explore this in simulation with the lower body of our humanoid robot CogIMon.

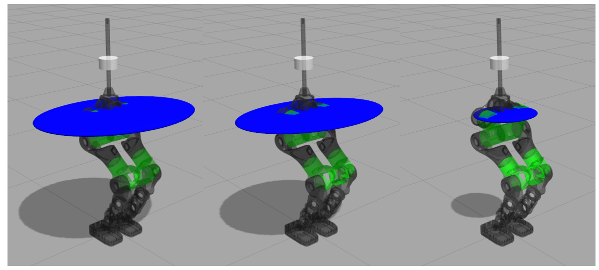
The lower body of CogIMon has 12 actuated DoF (6 for each leg: 3 hip joints, 1 knee joint and 2 ankle joints) [15]. A fake mass link has been fixed to the top of pelvis link to represent the upper body. In simulation, the robot has been command to a given posture (CoM height equals to $0.8m$). Manipulability corresponding to sub-matrices in (2) have been computed and listed in Table I. According to the data in the table, the contribution from floating base

TABLE I: Manipulability Contribution

Manipulability	ω	ω_b	ω_a	$\omega_a : \omega_b$
Linear momentum	209575.85	207226.92	61.40	1:3375
Angular momentum	66.03	42.73	6.63	1:6



(a) Linear momentum ellipsoids generated from $\boldsymbol{A}_l$, $\boldsymbol{A}_{lb}$ and $\boldsymbol{A}_{la}$



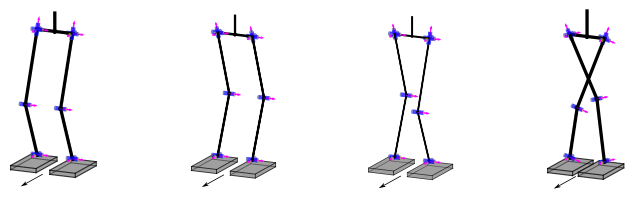
(b) Angular momentum ellipsoids generated from $\boldsymbol{A}_k$, $\boldsymbol{A}_{kb}$ and $\boldsymbol{A}_{ka}$

Fig. 2: Momentum manipulability ellipsoids. For better visualization, a scale factor 10^{-2} has been applied to those linear momentum ellipsoids. Because of the scale disparities between linear and angular momentum, a different scale factors 10^{-1} have been applied to angular momentum ellipsoids. Linear momentum ellipsoids have been plotted in red color and angular momentum ellipsoids in blue for differentiation.

velocity $\dot{\boldsymbol{q}}_b$ dominant the linear part ($\omega_{la} : \omega_{lb}$ = 1:3375). However this is not the case for angular momentum, actuated joint velocity $\dot{\boldsymbol{q}}_a$ contributed a comparable part ($\omega_{ka} : \omega_{kb}$ = 1:6) of angular momentum for the system. It is more straightforward to compare the contribution by observing the different manipulability ellipsoids as shown in Figure 2. All the results indicate that the actuated joint velocity $\dot{\boldsymbol{q}}_a$ has very limited contribution to the system linear momentum (or CoM velocity) but has a considerable amount of influence on the angular momentum. As a result, this paper will focus on studying how body movement contributes to the angular momentum of the system.

B. Angular Momentum Manipulability Related To Different Configurations

In this part, four different configurations as shown in Figure 3 are going to be examined. The *forward/forward* configuration is just like human with two knees bending forward. It is possible for a robot to bend its knees backward with proper mechanical design and which results in the *backward/backward* configuration. One analogy is the *elbow-up* and *elbow-down* configurations for a manipulator. A mix of the previous two leads to the *forward/backward* configuration. It can be further extended to a *twist* configuration by crossing step the left foot to the right side of the right foot.



(a) (b) (c) (d)

Fig. 3: Four configurations of humanoid robot (arrows indicate forward moving direction): (a) *forward/forward*; (b) *backward/backward*; (c) *forward/backward*; (d) *twist*. Here, *forward* and *backward* means knee configuration.

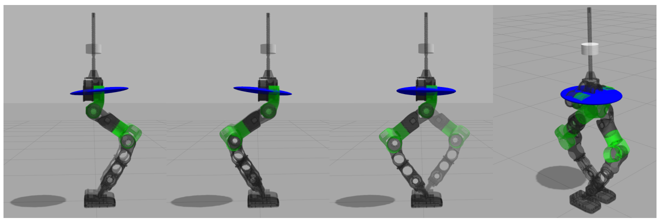


Fig. 4: Configuration related angular momentum ellipsoids of four different configurations from left to right: *forward/forward*, *backward/backward*, *forward/backward* and *twist*.

To evaluate the angular momentum manipulability of these four configurations, the feet of the robot are initiated at the same location on the ground (left foot and right foot has been swapped in *twist* configuration) and the CoM is regulated to the same position (x and y take the position of the center of the feet, $z = 0.8m$). The configuration related angular momentum ellipsoids are plotted in Figure 4 and corresponding manipulability indexes have been calculated and listed in Table II. It can be seen from the table that the configuration *forward/forward* and *backward/backward* have similar manipulability. However, these two configurations show different directional features (the first two plots in Figure 4). Considering the principle axes of the ellipsoid as the optimum direction to generate angular momentum, the two configurations have different optimum directions. Both *forward/backward* and *twist* configurations give better manipulability than single sided configuration (*forward/forward*, *backward/backward*). Among all the configurations, *forward/backward* gives the best angular momentum manipulability.

TABLE II: Angular Momentum Manipulability Related To Different Configurations

Manipulability	*f/f*	*b/b*	*f/b*	*twist*
ω_k	66.03	66.04	90.78	74.54
ω_{kb}	42.73	43.13	54.42	42.15
ω_{ka}	6.63	6.59	10.37	9.36

Note: *f*→*forward*, *b*→*backward*

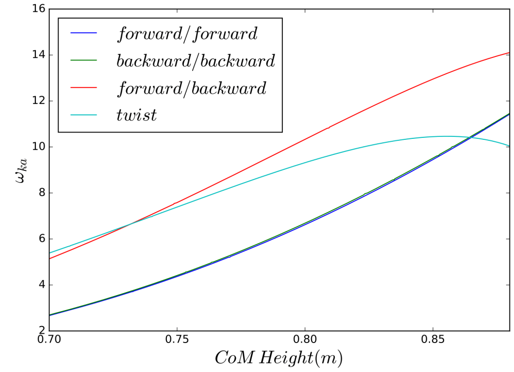


Fig. 5: Configuration related angular momentum manipulability through out the lift-up motion for four different configurations.

C. Lift-up Motion

In the previous section, it has been concluded that the *forward/backward* configuration gives the best angular momentum manipulability. However the result only valid for the specific posture for which the corresponding CoM height is $0.8m$. In this part, the manipulability is going to be examined for a series of configurations. The robot is requested to do a lift up motion, the CoM height has been commanded from $0.7m$ to $0.88m$ (due to leg length limit). Angular momentum manipulability index ω_{ka} value has been recorded through out the whole process. Results for all four configurations are shown in the Figure 5. It is obvious that the *forward/backward* is the best for this motion among the four configurations. The *twist* configuration shows good manipulability with lower CoM height, and it decreases as the robot lift-up. The other two configurations have no difference for this motion. One might notes that the angular momentum manipulability index increases as the robot lift-up, this relationship is reversed for the linear one which decreases as the robot lift-up. This can be interpreted in the way that as the masses of the robot distributed further and further away from the CoM, they gain more and more influences on the centroidal angular momentum with increasing moment arms.

In general, the proposed configurations *forward/backward* and *twist* give better angular momentum manipulability for a wide range of postures. This is not the only benefits it brings to the robot, they also enables the new movement possibility: cross-step.

III. Cross-step

Robust walking of humanoid robots often requires the robot to be adaptive to external disturbances and terrain irregularities. Three push recovery strategies that allow the robot to recover from different levels of external push has been presented in [16] . Push recovery stepping strategies have been proposed in multiple works [5] [17] [18] [19]. Additionally, different model predictive control (MPC) formulation have been proposed to updates foot placement

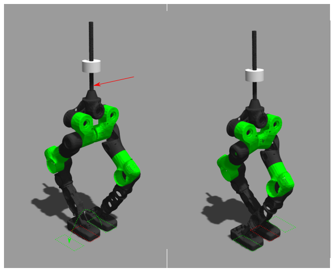


Fig. 6: Corss-step. The red arrow indicates a push force from left to right acting on the robot at a certain moment, and this initiates the cross-step action: the left foot swing over the right foot and lands on the right side of it.

online [20] [21] [22] [23]. To meet real-time requirement, linear models are often chosen as template model to perform iterative online optimization involved in the MPC control scheme. Non-linear formulation which involve step timing optimization have been explored in several recent studies [24] [25] [26]. Considering the worst case scenario in which the robot has been heavily pushed towards the right during the right support phase, a two step strategy is necessary: put down the left foot as close as possible to the right foot within as short as possible duration, followed a large right side step. For human, a more natural reaction would be cross their legs to make a cross step directly. This action is however risky or mechanically impossible for most existing robots. The proposed configurations *forward*/*backward* or *twist* could be a solution to this problem.

The cross-step action with *forward*/*backward* configured robot is shown in the Figure 6. As can be seen from the figure, the robot switches from *forward*/*backward* configuration to *twist* configuration with one cross step. Actually, the same can happen from *twist* configuration to *forward*/*backward* configuration. As a result, the robot can switch between these two configurations infinitely which means that the robot can do multiple cross-steps continuously as plotted in Figure 7. One might notice that self-collision happens between the hip-pitch links, this is due to the fact that the mechanical design is finished before we come up with this cross-step idea. But it is absolutely possible to avoid this problem with proper design.

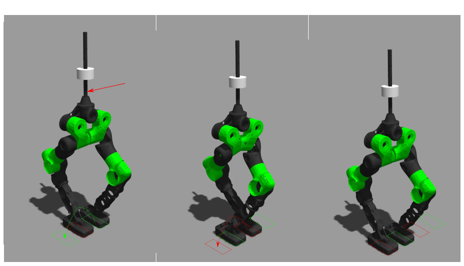
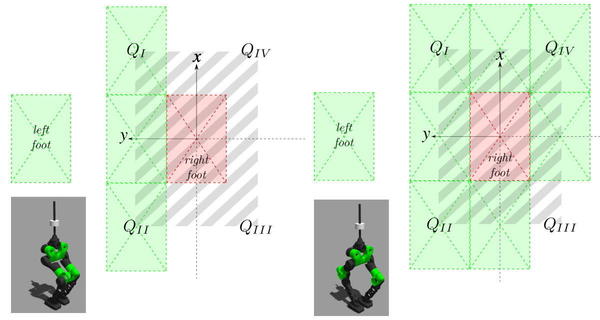


Fig. 7: Multiple corss-steps. The robot switches from *forward*/*backward* to *twist*, and then from *twist* to *forward*/*backward*. Footprints has been labeled with squares, the green ones represent the left footprints and the red ones are for the right foot.



(a) *forward*/*forward* (b) *twist*

Fig. 8: Design region for left swing foot while the robot takes different configuration. Assuming in right single support phase, the right stance foot has been plotted in red and a fixed frame has been attached to its center. The swing left foot is in green and several possible landing prints have been plotted for reference. Grey strip labels out unfeasible regions due to self-collisions between feet.

A. Cross-step Feasible Region

With the possibility to do cross-step action, the feasible region for the swing foot is enlarged. For footstep planning, the feasible region $\mathcal{F}$ of the swing foot is usually defined by:

$$\mathcal{F} \in \mathcal{D} \cap \mathcal{K} \cap \mathcal{C} \tag{8}$$

where $\mathcal{D}$ is the design region, $\mathcal{K}$ is the kinematic feasible region, $\mathcal{C}$ is the collision-free region. For the two cases illustrated in Figure 8, they have different design region $\mathcal{D}$:

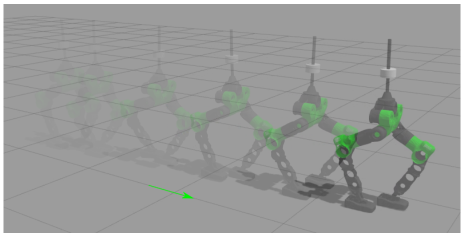
$$\mathcal{D}_{f/f} = \{(x, y) \in Q_I \cup Q_{II}\} \tag{9}$$

$$\mathcal{D}_{f/b} = \{(x, y) \in Q_I \cup Q_{II} \cup Q_{IV}\} \tag{10}$$

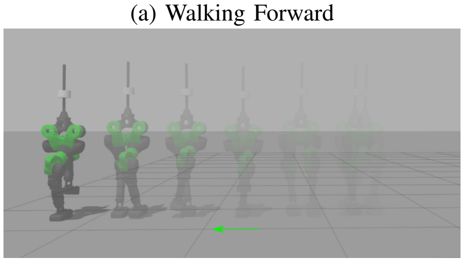
where Q_I, Q_{II} and Q_{IV} stands for quadrant I, II and IV of (x, y) plane, f/f and f/b stands for *forward*/*forward* and *forward*/*backward*. f/b configuration increased the design region with one more quadrant comparing to f/f configuration .

B. Simulation

The walking motion is simulated in Gazebo + ROS (Robot Operating System) environment. For Gazebo simulator, ODE physics engine has been chosen. Each joint of the robot could be controlled in position mode or torque controlled. In the simulation, we choose pure torque control mode for all the joints. Robot Odometry data (pelvis position and velocity) and joint states (positions and velocities) has been sent to control system as state feedbacks. A two level hierarchical control system has been used to generate walking motion for the robot. High-level controller plans Cartesian space trajectories for CoM and feet. The low-level controller is a whole-body controller which takes the desired CoM and feet trajectories as input and finds out the joint-torques for all joints. The controller has been formulated as a quadratic



(a) Walking Forward



(b) Walking Sideways

Fig. 9: Robot walking in different direction: (a) The robot is walking forward, (b) the robot is walking towards its right direction.

optimization problem whose goal is to track desired trajectories as good as possible and at the same time with respect to all kinds of constraints, such as dynamic feasibility, friction cone, torque limits [22] [27] [28]. The whole-body controller is running at a much higher frequency than the high level controller, in walking simulation 1000 Hz has been used.

C. Walking Planner

The model we used to generate the reference motion is linear inverted pendulum model (LIPM) and many walking pattern generation methods based on this model have been proposed [29] [30] [31] [32]. The model is composed of a point mass and a massless telescopic leg. Therefore, the planner based on this model provide no information about the configuration of the robot. It only generates Cartesian space reference trajectories and it replans these trajectories at support switching moment for next three steps. Here it is assumed that no double support phase exists, left support phase switches to right support phase simultaneously. The planner is formulated as a liner model predictive control problem which takes future foot placements as optimization variables [22] [23]. Suppose there is a desired average velocity $\boldsymbol{v}_{ref}$ the robot needs to track. The first optimization goal is to minimize the least square error between this desired velocity $\boldsymbol{v}_{ref}$ and future switching moment CoM velocities (three steps have been considered in this work). The other goal is to minimize the least square errors between replanned footsteps and desired footsteps. Desired footsteps are calculated based on desired velocity $\boldsymbol{v}_{ref}$ with consideration of inter-feet clearance to avoid self collision between feet. With cross-step enabled, the feasible region of desired footsteps expanded as shown in (b) of Figure 8. However the stance foot makes the feasible region non-convex, in this case, we have to select a convex sub-region to make problem convex and solvable. In this paper, convex sub-region has been chosen in a heuristic way: either $Q_I \cup Q_{II}$ or Q_{IV} depending on how much lateral velocity been commanded or how much lateral disturbance been applied. Taking the lateral velocity as example, a threshold value could be defined in advance, if commanded lateral velocity goes beyond this threshold, the design region switch from $Q_I \cup Q_{II}$ to Q_{IV}. The same rule applies for lateral disturbance case.

Walking motion in different directions have been simulated as shown in Figure 9. Walking forward and walking sideways have been demonstrated. The robot takes cross-step action in the sideways walking. Actually, the cross-step action is trigged by the reference velocity $\boldsymbol{v}_{ref}$. with a y component of $\boldsymbol{v}_{ref}$ beyond certain threshold, it will trigger the cross-step motion for the robot. A small one will results in small side step without crossing legs. A strong side push on the robot could also trigger the cross-step in the same direction. One thing worth noting is that the robot does not have to switch back to *forward*/*backward* configuration to be able to walk forward. That is to say, the robot can perform walking forward motion in *twist* configuration as well and it is shown in Figure 10. This guarantees the robot could change walking direction at any stage of cross-step.

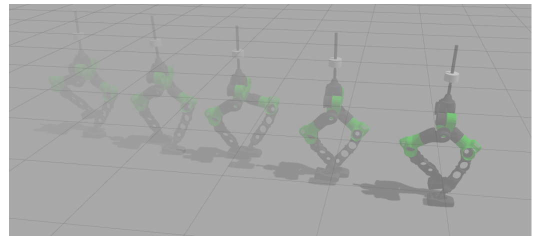


Fig. 10: Walking forward with *twist* configuration.

IV. CONCLUSION

In this paper, we propose two new configurations for humanoid balancing and walking. We have compared them with other regular configurations in terms of centroidal momentum manipulability. They indeed provide better angular momentum manipulability. One major benefit of the proposed configurations is that they enable the robot to do cross-step motion. This is a useful skill for humanoid push recovery but long being ignored due to hardware limitation. With cross-step enabled, the robot is more robust to lateral direction disturbances. However, due to non-convex feasible region, traditional convex optimization can not be directly applied to plan cross-step. A simple heuristic has been proposed to overcome this problem. Walking simulation has been performed successfully to verify the proposed cross-step idea.

ACKNOWLEDGMENT

This work is supported by European Community's Horizon 2020 robotics program CogIMon (ICT-23-2014, 644727).

REFERENCES

[1] M. Vukobratovic, A. Frank, and D. Juricic, "On the stability of biped locomotion," *IEEE Transactions on Biomedical Engineering*, no. 1, pp. 25–36, 1970.

[2] M. Vukobratović and B. Borovac, "Zero-moment pointthirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 01, pp. 157–173, 2004.

[3] A. Goswami, "Foot rotation indicator (fri) point: A new gait planning tool to evaluate postural stability of biped robots," in *IEEE International Conference on Robotics and Automation*, vol. 1, 1999, pp. 47–52.

[4] A. Goswami and V. Kallem, "Rate of change of angular momentum and balance maintenance of biped robots," in *IEEE International Conference on Robotics and Automation*, vol. 4, 2004, pp. 3785–3790.

[5] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *International Conference on Humanoid Robots*, 2006, pp. 200–207.

[6] T. Yoshikawa, "Manipulability of robotic mechanisms," *The international journal of Robotics Research*, vol. 4, no. 2, pp. 3–9, 1985.

[7] ——, "Dynamic manipulability of robot manipulators," *Transactions of the Society of Instrument and Control Engineers*, vol. 21, no. 9, pp. 970–975, 1985.

[8] A. Bowling and O. Khatib, "The dynamic capability equations: a new tool for analyzing robotic manipulator performance," *IEEE transactions on robotics*, vol. 21, no. 1, pp. 115–123, 2005.

[9] N. Naksuk and C. G. Lee, "Zero moment point manipulability ellipsoid," in *IEEE International Conference on Robotics and Automation*, 2006, pp. 1970–1975.

[10] ——, "Utilization of movement prioritization for whole-body humanoid robot trajectory generation," in *IEEE International Conference on Robotics and Automation*, 2005, pp. 1079–1084.

[11] S. Cotton, P. Fraisse, and A. P. Murray, "On the manipulability of the center of mass of humanoid robots: Application to design," in *ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2010, pp. 1259–1267.

[12] Y. Gu, C. G. Lee, and B. Yao, "Feasible center of mass dynamic manipulability of humanoid robots," in *IEEE International Conference on Robotics and Automation*, 2015, pp. 5082–5087.

[13] M. Azad, J. Babič, and M. Mistry, "Dynamic manipulability of the center of mass: A tool to study, analyse and measure physical ability of robots," in *IEEE International Conference on Robotics and Automation*, 2017, pp. 3484–3490.

[14] D. E. Orin and A. Goswami, "Centroidal momentum matrix of a humanoid robot: Structure and properties," *dynamics*, vol. 4, p. 6, 2008.

[15] C. Zhou and N. Tsagarakis, "On the comprehensive kinematics analysis of a humanoid parallel ankle mechanism," *Journal of Mechanisms and Robotics*, vol. 10, no. 5, p. 051015, 2018.

[16] B. Stephens, "Humanoid push recovery," in *7th IEEE-RAS International Conference on Humanoid Robots*, 2007, pp. 589–595.

[17] M. Morisawa, K. Harada, S. Kajita, K. Kaneko, J. Sola, E. Yoshida, N. Mansard, K. Yokoi, and J.-P. Laumond, "Reactive stepping to prevent falling for humanoids." in *International Conference on Humanoid Robots*, 2009, pp. 528–534.

[18] J. Urata, K. Nshiwaki, Y. Nakanishi, K. Okada, S. Kagami, and M. Inaba, "Online decision of foot placement using singular lq preview regulation," in *11th IEEE-RAS International Conference on Humanoid Robots*, 2011, pp. 13–18.

[19] J. A. Castano, Z. Li, C. Zhou, N. Tsagarakis, and D. Caldwell, "Dynamic and reactive walking for humanoid robots based on foot placement control," *International Journal of Humanoid Robotics*, vol. 13, no. 02, p. 1550041, 2016.

[20] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl, "Online walking motion generation with automatic footstep placement," *Advanced Robotics*, pp. 719–737, 2010.

[21] B. J. Stephens and C. G. Atkeson, "Push recovery by stepping for humanoid robots with force controlled joints," in *International Conference on Humanoid Robots*, 2010, pp. 52–59.

[22] S. Faraji, S. Pouya, C. G. Atkeson, and A. J. Ijspeert, "Versatile and robust 3d walking with a simulated humanoid robot (atlas): A model predictive control approach," in *IEEE International Conference on Robotics and Automation*, 2014, pp. 1943–1950.

[23] S. Feng, X. Xinjilefu, C. G. Atkeson, and J. Kim, "Robust dynamic walking using online foot step optimization," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2016, pp. 5373–5378.

[24] P. Kryczka, P. Kormushev, N. G. Tsagarakis, and D. G. Caldwell, "Online regeneration of bipedal walking gait pattern optimizing footstep placement and timing," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015, pp. 3352–3357.

[25] M. Khadiv, A. Herzog, S. A. A. Moosavian, and L. Righetti, "Step timing adjustment: A step toward generating robust gaits," in *IEEE-RAS 16th International Conference on Humanoid Robots*, 2016, pp. 35–42.

[26] W. Hu, I. Chatzinikolaidis, K. Yuan, and Z. Li, "Comparison study of nonlinear optimization of step durations and foot placement for dynamic walking," in *IEEE International Conference on Robotics and Automation*, 2018, pp. 433–439.

[27] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson, "Optimization based full body control for the atlas robot," in *14th IEEE-RAS International Conference on Humanoid Robots*, 2014, pp. 120–127.

[28] S. Xin, Y. You, C. Zhou, C. Fang, and N. Tsagarakis, "A torque-controlled humanoid robot riding on a two-wheeled mobile platform," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017, pp. 1435–1442.

[29] 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 *IEEE International Conference on Robotics and Automation*, vol. 3, 2003, pp. 1620–1626.

[30] P.-B. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," in *IEEE-RAS international conference on humanoid robots*, 2006.

[31] R. Tedrake, S. Kuindersma, R. Deits, and K. Miura, "A closed-form solution for real-time zmp gait generation and feedback stabilization," in *IEEE-RAS 15th International Conference on Humanoid Robots*, 2015, pp. 936–940.

[32] C. Zhou, X. Wang, Z. Li, and N. Tsagarakis, "Overview of gait synthesis for the humanoid coman," *Journal of Bionic Engineering*, vol. 14, no. 1, pp. 15–25, 2017.