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

Human-like ZMP Generator and Walking Stabilizer based on Divergent Component of Motion

Haitao Wang[1], Zhongyuan Tian[1], Wenbin Hu[1] and Mingguo Zhao[2]

***Abstract*— To realize natural and fast bipedal walking, we mimic the ZMP trajectory from human walking data based on the concept of Divergent Component of Motion (DCM). This paper presents an omnidirectional gait generator that is lightweight and universal. In addition, we integrate DCM tracking and foot placement adjustment into our control framework to ensure a stable walking motion and push recovery. The method proposed was validated in RoboCup2018 AdultSize soccer competition, where our robot WALKER+ walked on grass field and kept balance after the push from other robots.**

I. Introduction

Humanoid robots are built to live and play with people in the near future, especially in environments designed for humans. The locomotion of bipedal robot is a complicated problem for scientists and engineers. One of the most popular concepts in online gait generation is to simplify the robot as a reduced order model, such as Spring Loaded Inverted Pendulum (SLIP) [1] and Linear Inverted Pendulum (LIP) [2]. However, for those position controlled robot, zero moment point (ZMP) [3] is an important criteria, thus ZMP based LIP is much commonly used in planning to reduce the time of optimization with full-dynamics model. Kajita proposes the widely used preview control [4] method and Spatially Quantized Dynamics (SQD) [5] based pattern generator. Wieber et al. presents a model predictive controller (MPC) and quadratic programming based gait generator [6]. Tedrake propose a closed-form planner based on polynomial ZMP reference using linear quadratic regulator (LQR) [7].

Recently, people have a better understanding of the LIP by splitting its motion into two components. One part called Divergent Component of Motion (DCM) simplifies the LIP model into two orthogonal dynamics, where only the divergent component should be considered. DCM is firstly used in push recovery [8] (another name 'Capture Point' [9] is more famous in this problem), and then it is used in gait generator and controller by Takenaka [10]. Englsberger extends its concept into 3D and has a great contribution to the DCM based gait control method [11]. Hopkins et al. do a lot of work in time-varying DCM [12].

Based on DCM, push recovery in walking can be solved in many ways [13], [14]. Khadiv simplifies the nonlinear step-timing optimization with foot location at the same time [15].

*This work was supported by UBTECH Robotics, Inc.

[1]Department of Automation, Tsinghua University, Beijing 100084, China. wht17@mails.tsinghua.edu.cn

[2]Center for Brain-Inspired Computing Research (CBICR), Beijing Innovation Center for Future Chip, & Department of Automation, Tsinghua University, Beijing 100084, China. mgzhao@mail.tsinghua.edu.cn

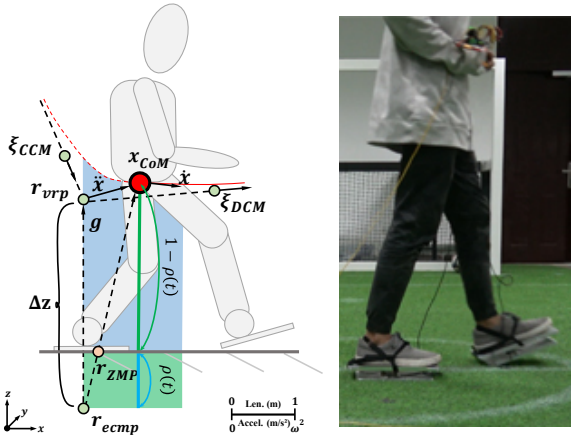


Fig. 1: Left: DCM dynamics on robot WALKER+. Right: Human walking experiment

Griffin utilizes QP to compute desired location when large disturbance occurs [16]. Jeong presents a CP feedback based walking stabilization [17]. Kamioka et al. simultaneously optimize the ZMP and footstep using analytical solution [18].

Most position controlled robots utilize impendance control using force torque sensors feedback, due to the fact that the required force in the dynamic model (LIP) can not be produced directly. Therefore, the gait generator is more important for those position controlled robot as a feed forward input. In the 1v1 competition at the RoboCup, real-time gait replanning is required, which means MPC based gait generation is too complicated. In this paper, we propose a polynomial spline based DCM generator using arbitrary ZMP key frame input, which includes the desired ZMP high-order derivative information. We also collect the human walking data to find the ZMP pattern and mimic the actions of human on our robot. At last, we present the control framework integrating DCM tracking controller and foot placement adjustment.

This paper is organized as follows: In section II: we review the dynamics of DCM and its features. The polynomial spline based DCM and ZMP generator and controller is given then. In section III: we introduce the ZMP mimicking system we use and shows the ZMP trajectory's feature. The walking disturbance stabilizer is presented in section IV. Section V shows the simulation and the hardware experiment results. Finally, we give a conclusion and suggest future work in section VI.

II. LIP BASED DCM GENERATOR AND CONTROLLER

A. Linear Inverted Pendulum and motion components

In this section, we briefly summrize some properties of 3D-LIP and its two motion components. To linearlize the Inverted Pendulum's dynamics, Kajita utilizes a constant height constraint [4], and based on it, Englsberger presents the Virtual Repellent Point (VRP) eliminating the effects of gravity in the nonlinear dynamics [11].

The key point in the LIP is to establish a linear mapping from the vector of distance to the vector of acceleration by

$$\ddot{\boldsymbol{x}} = \omega^2(\boldsymbol{x} - \boldsymbol{r}_{vrp}) \tag{1}$$

where $\boldsymbol{x} = [x_{com}, y_{com}, z_{com}]^T$ is the center of mass (CoM) position, and $\boldsymbol{r}_{vrp} = [x_{vrp}, y_{vrp}, z_{vrp}]^T$ is the position vector of VRP. The scalar of the mapping is denoted by $\omega^2 = g/\Delta z$, and the enhanced Centroidal Moment Pivot (eCMP) is located directly below the VRP, satisfying $\boldsymbol{r}_{ecmp} = \boldsymbol{r}_{vrp} - [0, 0, \Delta z]^T$.

Rewrite the dynamics of LIP using state equation, a linear state transformation can be found in the process of *eigenvector decomposition* [8], [18]. And the mean vector of the two new independent basis vectors equals to CoM.

$$\begin{bmatrix} \boldsymbol{\xi}_{DCM} \\ \boldsymbol{\xi}_{CCM} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x} + \dot{\boldsymbol{x}}/\omega \\ \boldsymbol{x} - \dot{\boldsymbol{x}}/\omega \end{bmatrix} \tag{2}$$

The decoupled dynamics (3) shows that the motion of LIP ($\boldsymbol{x}$) has two orthogonal components, $\boldsymbol{\xi}_{DCM}$ is unstable and diverging away from the $\boldsymbol{r}_{vrp}$, but another component $\boldsymbol{\xi}_{CCM}$ convergents to the $\boldsymbol{r}_{vrp}$.

$$\frac{d}{dt}\begin{bmatrix} \boldsymbol{\xi}_{DCM} \\ \boldsymbol{\xi}_{CCM} \end{bmatrix} = \begin{bmatrix} \omega & 0 \\ 0 & -\omega \end{bmatrix}\begin{bmatrix} \boldsymbol{\xi}_{DCM} \\ \boldsymbol{\xi}_{CCM} \end{bmatrix} + \begin{bmatrix} -\omega \\ \omega \end{bmatrix}\boldsymbol{r}_{vrp} \tag{3}$$

Therefore, if the unstable part ($\boldsymbol{\xi}_{DCM}$) is undercontrolled by moving VRP, whole motion of the CoM can be easily achieved. Englsberger explains this idea using serial decomposition, that VRP pushes away the DCM, but CoM follows the DCM [11].

B. Weighted ZMP Smoother

It is well known that $\boldsymbol{r}_{ZMP} = [p_x, p_y]^T$ can be calculated in the Inverted Pendulum model (take p_x for example).

$$p_x = x_{com} - \frac{z_{com}}{\ddot{z}_{com} + g}\ddot{x}_{com} \tag{4}$$

To set the relationship between ZMP and VRP, we can define a weight ratio $\rho(t)$ to recalculate r_{ZMP} in Fig. 1, which satisfies $\boldsymbol{r}_{ZMP} = \rho(t)\boldsymbol{x} + (1 - \rho(t))\boldsymbol{r}_{ecmp}$, and $\rho(t)$ can be solved according to the proportional relationship along the vertical green and blue line ($p_z = z_{ground}$).

$$\begin{aligned} p_x &= \rho(x_{com} - x_{vrp}) + x_{vrp} \\ \rho(t) &= \frac{z_{vrp} - \Delta z - z_{ground}}{z_{vrp} - \Delta z - z_{com}} \end{aligned} \tag{5}$$

It is a normal case that $z_{vrp} - \Delta z \neq z_{ground} \rightarrow \rho(t) \neq 0$, which means we can realize a Heel-to-Toe ZMP trajectory while the $\boldsymbol{r}_{vrp}$ trajectory remaining fixed in each step.

The supporting area is important for the ZMP based walking generator: for the robot with narrow feet, constant CoM's Height with $\rho = 0$ is more common, but for our big-foot robot WALKER+, we allow $\rho(t) \in [0, 1)$ to make the $\boldsymbol{r}_{ZMP}$ moves in the front of the projection of the VRP.

On the other hand, the drastic change of ZMP in Double Support Phase (DSP) will make it hard to minimize its tracking error. Using preview control is a good way to generate a smoother ZMP trajectory [4], but tuning the $\rho(t)$ is a much easier way to do it.

C. Generation of DCM and ZMP

There are many works on ZMP based CoM generation for bipedal robots, some are LQR and preview control based and need more computation [4], [6], [7]. Recently DCM based trajectory simplifies the calculation using piecewise polynomial DCM and ZMP [19], [18].

In our omnidirectional gait generator, we define foot-center point as the middle of two feet while robot standing. Then we generate the position and orientation of footprint in next N steps by command message $[\Delta L_{forward}, \Delta L_{side}, \Delta\theta]$ compatible with speed mode.

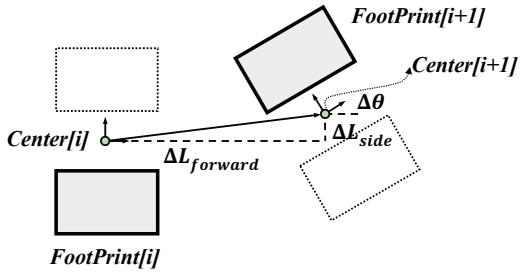


Fig. 2: Footprint Generation

To realize a simple planning, we use a polynomial spline to interpolate k_i DCM points each step period from current state to a virtual final state after N steps. The idea of the key DCM point is from the Heel-to-Toe DCM planning in [11].

Here we explain the relationship in continuity between DCM and ZMP in our method. From equation (3), we can derive ($\boldsymbol{\xi}_{DCM}$ is denoted by $\boldsymbol{\xi}$ and $\boldsymbol{r}_{vrp}$ is denoted by $\boldsymbol{v}$) .

$$\boldsymbol{v} = \boldsymbol{\xi} - \frac{1}{\omega}\dot{\boldsymbol{\xi}} \tag{6}$$

If we use polynomial spline to connect two states, the VRP will keep continuous except for the start and the end of transition in high-order level. We denote vector $[\boldsymbol{\xi}, \dot{\boldsymbol{\xi}}, ..., \boldsymbol{\xi}^{(m)}]$ as $\boldsymbol{\Xi}^{[m]}$, and the subscript of $\boldsymbol{\Xi}^{[m]}$ indicates the order of the footprints. Thus $\boldsymbol{\Xi}_{i-1}^{[m]}$ and $\boldsymbol{\Xi}_{i}^{[m]}$ can determine the polynomial order as $(2m+1)$ between footprint $i-1$ and i.

As the state $\boldsymbol{\Xi}_{i}^{[m]}$ is used in two polynomials that $\boldsymbol{\Xi}_{i-1}^{[m]} \rightarrow \boldsymbol{\Xi}_{i}^{[m]}$ and $\boldsymbol{\Xi}_{i}^{[m]} \rightarrow \boldsymbol{\Xi}_{i+1}^{[m]}$, we denote the vector $[\boldsymbol{v}, \dot{\boldsymbol{v}}, ..., \boldsymbol{v}^{(m)}]$ as $\boldsymbol{V}$, then $\boldsymbol{V}_i$ can be calculated using $\boldsymbol{\Xi}_{i}^{[m]}$ with polynomial

parameters. The continuity from $\boldsymbol{v}$ to $\boldsymbol{v}^{(m-1)}$ is guaranteed between two transitions, only the $\boldsymbol{v}^{(m)}$ is not continuous.

Actually, we cannot select the $\boldsymbol{\Xi}^{[m]}$ easily without the help of the VRP, which can be generated much more directly from the footprints.

We define $t_{i,j}$ as the time duration of DCM transition from $(j-1)^{th}$ key point $\boldsymbol{\Xi}^{[m]}_{i,j-1}$ to j^{th} point $\boldsymbol{\Xi}^{[m]}_{i,j}$ in the i^{th} step. $(i = 1, 2, ..., N, j = 1, 2, ..., k_i)$. We initialize these key DCM point using discrete input $vrp_{i,j}$ based on backward iteration [11]. We can get some broken lines connected by $\boldsymbol{\xi}_{i,j}$, the time duration $T_{i,j}$ from $\boldsymbol{\xi}_{i,j} \rightarrow \boldsymbol{\xi}_{i,j+1}$ can be calculated as follows, $\alpha_{i,j}$ are weight parameters.

$$T_{i,j} = \begin{cases} (1-\alpha_{i,j})t_{i,j} & +\alpha_{i,j+1}t_{i,j+1}, 1 \leq j < k_i \\ (1-\alpha_{i,j})t_{i,j} & +\alpha_{i+1,1}t_{i+1,1}, j = k_i \end{cases} \tag{7}$$

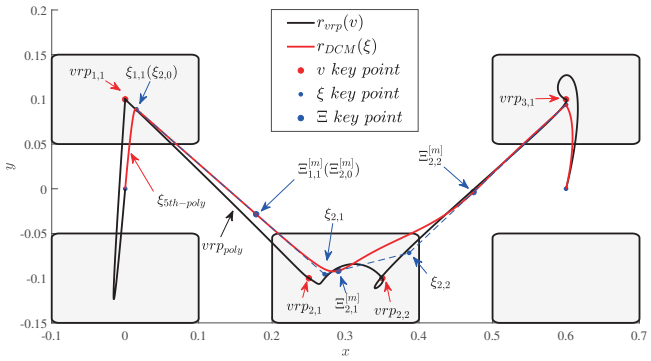


Fig. 3: DCM and VRP Generation

After the backward iteration, we suppose VRP is const first so as to use

$$\boldsymbol{\xi}(t) = (\boldsymbol{\xi}(0) - \boldsymbol{v}(0))e^{\omega t} + \boldsymbol{v}(0) \tag{8}$$

to calculate a reference DCM position (the first variable in the vector $\boldsymbol{\Xi}^{[m]}_{i,j}$).

$$\boldsymbol{\Xi}^{(0)}_{i,j} = (\boldsymbol{\xi}_{i,j} - \boldsymbol{v}_{i,j})e^{\omega T_{i,j}} + \boldsymbol{v}_{i,j} \tag{9}$$

Then we differentiate on both sides of the equation (3) for $(m-1)$ times.

$$\begin{aligned} \boldsymbol{\xi}^{(1)} &= \omega(\boldsymbol{\xi} - \boldsymbol{v}) \\ \boldsymbol{\xi}^{(2)} &= \omega(\boldsymbol{\xi}^{(1)} - \boldsymbol{v}^{(1)}) \\ &... \\ \boldsymbol{\xi}^{(m)} &= \omega(\boldsymbol{\xi}^{(m-1)} - \boldsymbol{v}^{(m-1)}) \end{aligned} \tag{10}$$

The DCM key point vector $\boldsymbol{\Xi}^{[m]}$ is determined by current DCM position and $\boldsymbol{V}^{[m-1]}$, which means we can set the feature of VRP trajectory as we want using its higher derivative information.

The formulation during i^{th} step from $\boldsymbol{\Xi}_{i-1,k_i}(\boldsymbol{\Xi}_{i,0})$ to $\boldsymbol{\Xi}_{i,k_i}(\boldsymbol{\Xi}_{i+1,0})$ can be defined by piecewise polynomial:

$$\begin{aligned} \boldsymbol{\xi}_i(t) &= \sum_{j=1}^{k_j} h_j(t) \sum_{s=0}^{2m+1} \beta_{j,s} t^s, \\ h_j(t) &= \begin{cases} 1, \sum_{s=1}^{j-1} t_{i,s} & \leq t < \sum_{s=1}^{j} t_{i,s} \\ 0, else \end{cases} \end{aligned} \tag{11}$$

where $\beta_{j,s}$ are polynomial parameters.

Then we can draw DCM and VRP trajectory in Fig. 3, according to equation (6).

Due to the fact that CoM follows the DCM trajectory, with a stable first-order dynamics:

$$\dot{\boldsymbol{x}} = -\omega(\boldsymbol{x} - \boldsymbol{\xi}) \tag{12}$$

We can obtain a CoM's iterative expression where Δt is the control period of the gait:

$$\boldsymbol{x}(t + \Delta t) = e^{-\omega \Delta t}(\boldsymbol{x}(t) - \boldsymbol{\xi}(t)) + \boldsymbol{\xi}(t) \tag{13}$$

Note: a) The difference between Englsberger's [19] and our method is that we consider all derivatives of $\boldsymbol{r}_{vrp}$ in non-zero conditions, therefore we can generate a much smoother trajectory for both ZMP and CoM. b) We choose the polynomial order as 5, and two key VRP&DCM points in 2^{nd} step in Fig.3. But in the following experiment shown in this paper, we choose each degree of the polynomial as 3, and the number of key points for one step equals to 4 in whole step period (DSP and SSP).

D. ZMP and DCM controller

We utilize DCM tracking controller and position-based ZMP controller in our system. The control law used in DCM tracking is:

$$\begin{aligned} \boldsymbol{v}_{des} &= \boldsymbol{v}_{ref} + (1 + \frac{K_p}{\omega})\boldsymbol{\xi}_{err} + K_d(\dot{\boldsymbol{\xi}}_{err}) \\ \boldsymbol{\xi}_{err} &= \boldsymbol{\xi}_{est} - \boldsymbol{\xi}_{ref} \end{aligned} \tag{14}$$

And the position-based ZMP stabilizer use the law:

$$\ddot{\boldsymbol{x}}_{des} = k_f \omega^2 (\boldsymbol{v}_{est} - \boldsymbol{v}_{des}) + \ddot{\boldsymbol{x}}_{ref} \tag{15}$$

VRP can be estimate by equation (5):

$$\begin{aligned} \boldsymbol{v}_{x,est} &= \frac{1}{1-\rho(t)}(p_x - \rho(t)\boldsymbol{x}_{x,est}) \\ \boldsymbol{v}_{y,est} &= \frac{1}{1-\rho(t)}(p_y - \rho(t)\boldsymbol{x}_{y,est}) \\ \boldsymbol{v}_{z,est} &= \Delta z + \frac{1}{1-\rho(t)}(z_{ground} - \rho(t)\boldsymbol{x}_{z,est}) \end{aligned} \tag{16}$$

The whole control system we use will be explained in section V.

III. Mimicking Human ZMP Trajectory

One of the main ideas of this paper is to design desired ZMP trajectories mimicking human walking data. We have proposed a polynomial spline based gait generator in section II. In this section, we will briefly introduce the process of extracting human ZMP data and mimicking walking pattern of human. Harada did some similar job in HRP-4C [20].

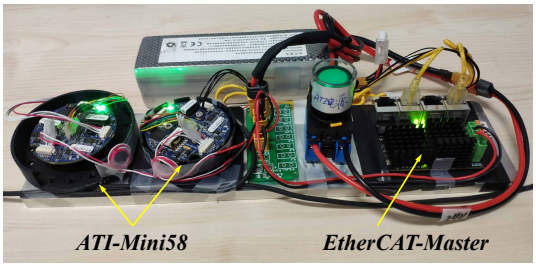
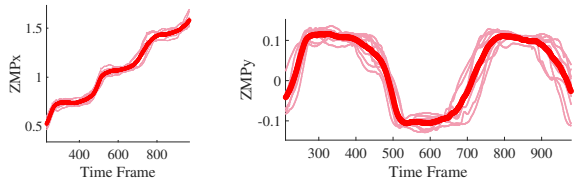


Fig. 4: ZMP Capture Device

We use the same 6-axis force-torque sensors as those on the robot, including the same sole structure. Ten fixed footprints are drawn on the floor in advance, while the participants only need to wear the special shoes to walk along the preplanned path. Meanwhile, another motion capture system is used to capture the swing foot trajectory, which is not modeled in reduced-order model we utilize.



(a) ZMPx (b) ZMPy

Fig. 5: Human Walking ZMP Stream

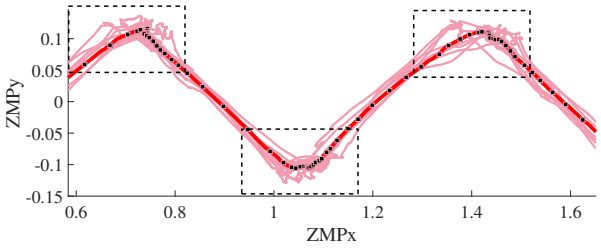


Fig. 6: ZMP Trajectory and Key Frame

Fig. 5 and Fig. 6 shows the pattern of human walking after data processing (scaling and intercepting). The dark red line is the average ZMP data from 9 experiments. And the light pink lines represent the origin data, the pattern of ZMP_x and ZMP_y is significant in the Fig. 6.

We decided to use the filterd ZMP data to get the key frame directly, for we just use the $\boldsymbol{v}, \boldsymbol{v}^{(1)}, ..., \boldsymbol{v}^{(m-1)}$ in gait planning to achieve a similar ZMP trajectory in Fig. 3.

IV. Walking Disturbance Stabilizer

We already have the DCM tracking controller and ZMP stabilizer, but robot still falls when being pushed or walking on uneven terrain. Thus we propose a stabilizer to replanning DCM and adjust the posture of the landing foot.

A. DCM Replanning

Considering the heavy legs our robot has, step adjustment cannot be so fast. We add an offset on the location of footprints at the end of each step, through that we can affect the planning of VRP and DCM directly.

$$\Delta \boldsymbol{x}_{footprint} = \boldsymbol{\xi}_{est,T} - \boldsymbol{\xi}_{ref,T} \tag{17}$$

The modification can be observed in Fig. 7, the robot is stepping in place when being pushed in one direction, and the periodic trajectories of VRP and DCM move in next step to adapt that push.

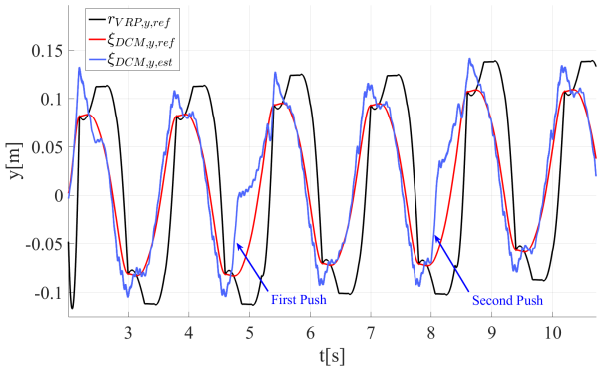


Fig. 7: Gait Replanning

B. Foot Landing Stabilizer

In order to reduce the impact of the landing, we add admittance control into the ankle joint, and we also change the swing foot trajectory 10 times a step by adjusting the desired landing posture.

We define a transform function used as a weight ratio between old and new spline trajectory.

$$\begin{aligned} \boldsymbol{x}_{swing}(t) &= \boldsymbol{x}_{old}(t)\left(1 - f_{tran}(t')\right) + \boldsymbol{x}_{new}(t) f_{tran}(t') \\ f_{tran}(t') &= 6{t'}^5 - 15{t'}^4 + 10{t'}^3 \\ t' &= \frac{t - t_0}{t_1 - t_0}, t_0 \le t \le t_1 \end{aligned} \tag{18}$$

The landing posture is adjusted according to [21], we extended its method by adding sagittal adjustment. Assume that the stance foot must have at least one point connected to the ground, and the error of the joint angle is small enough, then we can estimate the posture of the stance foot through IMU data and forward kinematics.

We use a Motion Capture based swing foot posture (position and pose) trajectory generater to get a new polynomial function $\boldsymbol{x}_{new}(t)$. Finally, the desired swing foot trajectory is calculated by equation (18).

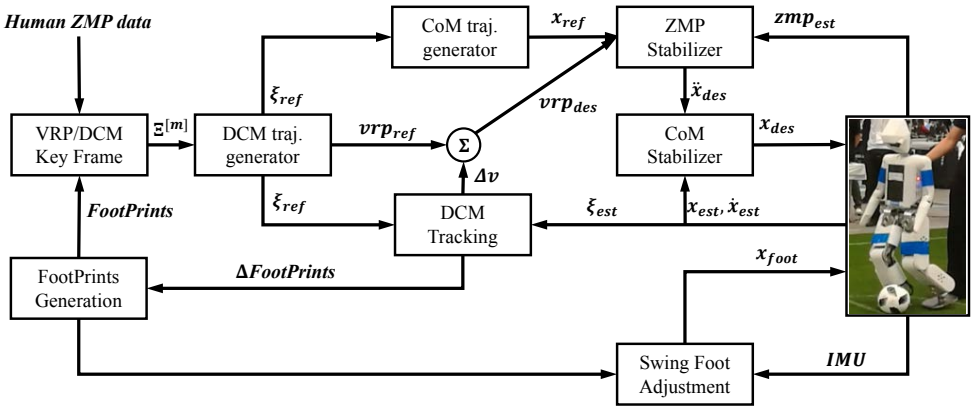


Fig. 8: Overall Control System of WALKER+

V. Experiment Results

To evaluate the proposed human-like gait generator and walking stabilizer, we performed numerous simulations of the humanoid robot WALKER+ in Simscape of MATLAB, the hardware experiments were conducted on the 50kg robot with real-time control system for the RoboCup competition games [22].

A. DCM Tracking Simulation

Fig. 9 shows two step response with different PD parameters for DCM tracking system (14) in MATLAB simulation. The desired DCM shifts to 0.06m in y direction at 0.0s in simulation time, after 5.0s it shifts to zero back. The response time of lower P gain group is 0.9s, and another group with higher P gain achieves the desired DCM in 0.4s, but there seems to be more oscillation. Therefore, setting a high proportional gain properly in the DCM tracking controller can improve the response rate of whole gait controller.

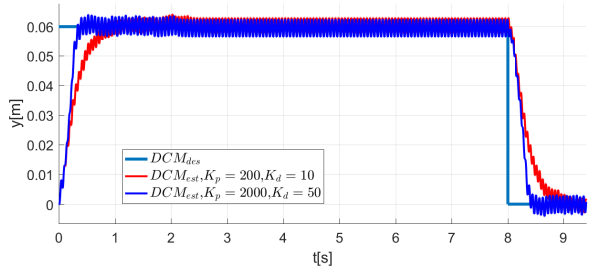


Fig. 9: DCM tracking Simulation

In this paper, we set $K_p = 2000, K_d = 50$ in simulation and $K_p = 500, K_d = 20$ in the hardware experiment to track the DCM of the robot.

B. Human-like Walking with Control

As a result of using human walking data to design a proper VRP key frame, we realize a stable walking with large steps both in simulation and hardware. The leg length of our robot is about 0.7m and the footprint length is 0.22m, we achieve 0.6m/step in simulation and 0.4m/step on hardware experiment, with $T_{step} = 1.1s, DSP = 25\%$. Fig. 10 shows the snapshot of walking in 4 fps.

We have tried Heel-to-Toe (HT) strategy in our gait generator to extend the step length, but we soon find that a long distance of HT will affect the performance of DCM tracking. In Fig. 6, we also find the distance of HT is not significant in SSP. The DSP of Human data is about 40%, and the range of HT is $\Delta X = 0.07m, \Delta Y = 0.03m$.

In Fig. 10, the robot ZMP planner successfully imitates the walking of human, what's more, in the 1-vs-1 soccer game at the RoboCup, the real-time human-like gait generator and walking stabilizer played an important role in the process of our robot winning the 3^{rd} place.

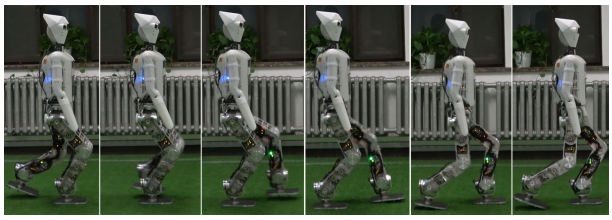


Fig. 10: Hardware experiment snapshot of walking

C. Push Recovery

The overall control system in Fig. 8 can break down the impact into several parts: a) When large disturbance happens, robot will adjust the previewed footprints to replan the gait according to $\Delta\xi$ before a new step period. b) DCM tracking controller will compensate the error caused by reduced order model and joint position lag. c) For our robot is position controlled, impendance control and swing foot adjustment are used to decrease the impact of foot landing and upper body inclination.

In Fig. 7, our robot can resist the impact of $140N \times 0.1s$. And the hardware test in the RoboCup Technical Challenge shows the robot can keep stepping after being hit by a 3Kg bag released from a 0.9m height circular orbit.

VI. CONCLUSION

A. HT like Human

One of the main contributions in this paper is to build a complete human-like gait generation and control system for bipedal robot based on Divergent Component of Motion. As long as the VRP key point position and desired high-order derivatives are designed, we can generate DCM and CoM trajectories through VRP key points in closed form.

The Heel-to-Toe strategy can not be used arbitrarily to extend the step length, therefore mimicking the human's ZMP data is effective in our polynomial spline based method.

B. Walking Stabilizer on Grass Field

Our robot had an excellent performance in RoboCup 2018 Competetion, it overcame the odds (soft grass field, other robot pushing and long time walking) and behaved like a real strong man when shooting and blocking the ball. The walking stabilizer is the key for the robot to keep balance when disturbance occurs.

C. Future Work

- We currently use the human data of ZMP directly, for the dataset is quite small. We will continue to collect human walking data and build an open source project. What we want to learn is the number of patterns during human walking with robot feet. The final goal is to realize an online walking learning on the dynamic features like ZMP or GRF.
- The proposed method is actually a 3-D gait generation, thus we will try to mimicking human climbing stairs in the near future. However, walk upstairs dynamically requires more torque output for the joint, on the one hand, more powerful joint unit is being designed, on the other hand, straight-knee walking and climbing algorithm should be considered.

ACKNOWLEDGMENT

The authors want to thank the UBTECH's staff from Beijing Research Institute for providing the humanoid robot WALKER+, which is real-time controlled using EtherCAT communication protocol at 1kHz.

This work was partially supported by UBTECH Robotics, Inc. and Tsinghua University.

We are also grateful to the anonymous reviewers for their detailed suggestions.

REFERENCES

[1] R. Blickhan, "The spring-mass model for running and hopping," *Journal of biomechanics*, vol. 22, no. 11-12, pp. 1217–1227, 1989.

[2] S. Kajita and K. Tani, "Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*. IEEE, 1991, pp. 1405–1411.

[3] M. Vukobratovic and D. Juricic, "Contribution to the synthesis of biped gait," *IEEE Transactions on Biomedical Engineering*, no. 1, pp. 1–6, 1969.

[4] 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 *ICRA*, vol. 3, 2003, pp. 1620–1626.

[5] S. Kajita, M. Benallegue, R. Cisneros, T. Sakaguchi, S. Nakaoka, M. Morisawa, K. Kaneko, and F. Kanehiro, "Biped walking pattern generation based on spatially quantized dynamics," in *Humanoid Robotics (Humanoids), 2017 IEEE-RAS 17th International Conference on*. IEEE, 2017, pp. 599–605.

[6] P.-b. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," in *Humanoid Robots, 2006 6th IEEE-RAS International Conference on*. IEEE, 2006, pp. 137–142.

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

[8] T. Koolen, T. De Boer, J. Rebula, A. Goswami, and J. Pratt, "Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models," *The International Journal of Robotics Research*, vol. 31, no. 9, pp. 1094–1113, 2012.

[9] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *Humanoid Robots, 2006 6th IEEE-RAS International Conference on*. IEEE, 2006, pp. 200–207.

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

[11] J. Englsberger, C. Ott, and A. Albu-Schäffer, "Three-dimensional bipedal walking control based on divergent component of motion," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 355–368, 2015.

[12] M. A. Hopkins, D. W. Hong, and A. Leonessa, "Humanoid locomotion on uneven terrain using the time-varying divergent component of motion," in *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*. IEEE, 2014, pp. 266–272.

[13] H. Wang and M. Zhao, "A robust biped gait controller using step timing optimization with fixed footprint constraints," in *Robotics and Biomimetics (ROBIO), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1787–1792.

[14] Y. Kang, Z. Tian, and M. Zhao, "The icp-based controllability criterion and its application to bipedal walking control," in *Robotics and Biomimetics (ROBIO), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2425–2432.

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

[16] R. J. Griffin, A. Leonessa, and A. Asbeck, "Disturbance compensation and step optimization for push recovery," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 5385–5390.

[17] H. Jeong, O. Sim, H. Bae, K. Lee, J. Oh, and J.-H. Oh, "Biped walking stabilization based on foot placement control using capture point feedback," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 5263–5269.

[18] T. Kamioka, H. Kaneko, T. Takenaka, and T. Yoshiike, "Simultaneous optimization of zmp and footsteps based on the analytical solution of divergent component of motion," in *Proceedings. 2018 IEEE International Conference on Robotics and Automation*. IEEE, 2018, pp. 1763–1770.

[19] J. Englsberger, G. Mesesan, and C. Ott, "Smooth trajectory generation and push-recovery based on divergent component of motion," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 4560–4567.

[20] K. Harada, K. Miura, M. Morisawa, K. Kaneko, S. Nakaoka, F. Kanehiro, T. Tsuji, and S. Kajita, "Toward human-like walking pattern generator," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 1071–1077.

[21] K. Hashimoto, K. Hattori, T. Otani, H.-O. Lim, and A. Takanishi, "Foot placement modification for a biped humanoid robot with narrow feet," *The Scientific World Journal*, vol. 2014, 2014.

[22] Z. Mingguo, W. Haitao, Z. Rongge, Z. Xueheng, T. Zhongyuan, W. Qilun, and H. Wenbin, "Tsinghua hephaestus 2018 adultsize team description," 2018.