Running Trajectory Generation Including Gait Transition between Walking Based on the Time-Varying Linear Inverted Pendulum Mode

Satomi Hanasaki†, Yuichi Tazaki†, Hikaru Nagano†, Yasuyoshi Yokokohji†

***Abstract*— This paper presents a novel running trajectory generation method for humanoid robots based on the time-varying linear inverted pendulum mode (LIPM). Vertical motion of the CoM, which is crucial for both steady-state running and transitions between walking and running, can be generated by simply optimizing the stiffness parameter of the LIPM during each contact phase. Since our method is a natural extension of walking trajectory generation method based on the conventional LIPM, it is capable of realizing natural and seamless gait transition between walking and running. Moreover, since the proposed method makes use of closed-form solutions of the LIPM, it is more computationally efficient than existing methods based on time-discretization with a fixed time step. Several simulations are performed to evaluate the efficiency of our method.**

I. Introduction

Humanoid robots are thought to be suitable for working in environment made for our everyday life, and developing humanoid robots with sufficient ability to move around and execute complicated tasks is regarded as one of the challenging goals of robotics.

For past many years, walking trajectory generation of humanoid robots has been studied. Kajita, et al. [1] developed the linear inverted pendulum mode (LIPM) for walking trajectory generation. Since of its simplicity, the LIPM was widely used in later influential studies ([2][3][4]).

There is another framework for humanoid motion planning. The SLIP, Spring Loaded Inverted Pendulum model, is designed primarily for running trajectory generation. Since Blickhan and Full [5] showed that the SLIP model can produce running trajectory of animals including human in high similarity, this model has been widely used in later studies related to running robots (e.g., [6][7][8]). The conventional SLIP model is two dimensional, while the 3D-SLIP model is proposed by Wensing and Orin.[9].

Bipedal robots should be able to make smooth transition between walking and running to realize high mobility in everyday life situation. Transition between walking and running has been discussed in some previous studies, where some of them utilize the LIPM while others utilize the SLIP model. Many studies of running trajectory generation based on the LIPM commonly focus on how to generate vertical CoM trajectories, because the original LIPM assumes that the CoM maintains a constant height from the ground. Early attempts realized running trajectory generation by synthesizing vertical and lateral motions separately ([10][11]). These methods need pre-planning of vertical motion or checking if the synthesized motion is physically feasible. Recent studies extended the conventional LIPM to generate various kinds of motion. Kamioka et al. [12] proposed a time-variant LIPM and realized gait transition between walking and hopping in experiments. Sugihara et al. [13] proposed gait transition by manipulation of the ZMP in 3D support region. This method is based on one-step look-ahead where the instantaneous capture point (ICP) is required to coincide with the landing point to ensure stability. Under this condition, however, faster running where ICP may move beyond the support region cannot be generated. The SLIP model based gait transition is also studied. Liu et al. [14] proposed a 3D Dual-SLIP model to express double support phase of walking and evaluated it in simulation. Because of the nonlinearlity of the SLIP model, walking trajectory generation based on the SLIP model tends to require high computation cost. Shahbazi et al. [15] derived an approximate solution of the 2D-SLIP model for walking, running and gait change, but approximation error could be large in specific condition. Therefore, more computationally efficient and physically accurate trajectory generation method which integrates walking and running is demanded.

The main contribution of this paper is to propose a novel trajectory generation method to realize vertical motion for steady-state running and transition between walking and running based on the time-varying LIPM. In the first step, values of the stiffness parameter of the LIPM for steady-state walking, running, and transition phases are computed by solving separate optimization problems with different boundary conditions. In the second step, a continuous-time trajectory of the CoM that is consistent with given footsteps and step durations is generated. Here, different stiffness values are assigned to each contact phase in order to generate appropriate vertical movement of the CoM. Both steps make full use of the closed-form solution of the LIPM for computational efficiency. To the authors' knowledge, our planner is the first running planner that makes use of the closed-form solution of the LIPM to generate trajectories of the CoM in 3D space. The validity of generated trajectories is tested in simulation experiments using a model of a life-sized humanoid robot.

II. Trajectory Generation Strategy

A. Structure of the planner

The flow of the proposed planning method is shown in Fig. 1. First, the nominal height (CoM height at landing and jumping) and the duration of each phase are inputted to the

†The authors is with Faculty of Engineering, Department of Mechanical Engineering, Kobe University, 1-1 Rokkodai, Nada-ku, Kobe, Japan. Email: hanasaki_satomi@stu.kobe-u.ac.jp, {tazaki, nagano, yokokohji}@mech.kobe-u.ac.jp

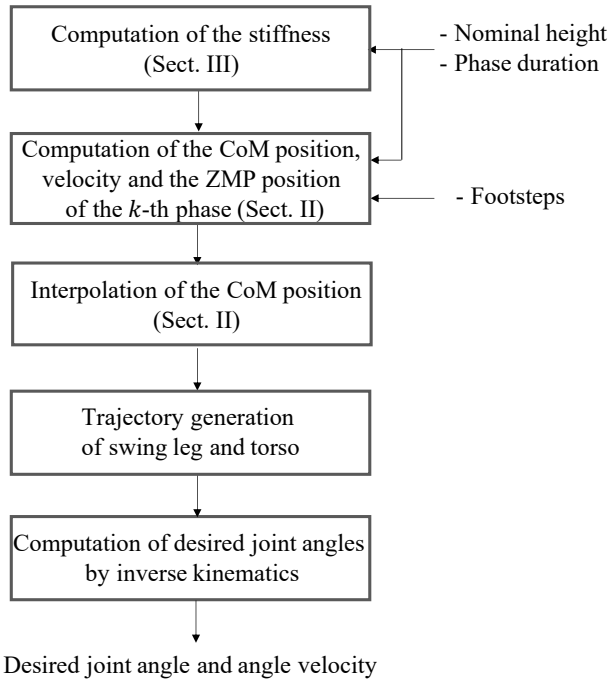

Fig. 1. Flow of the proposed motion planner

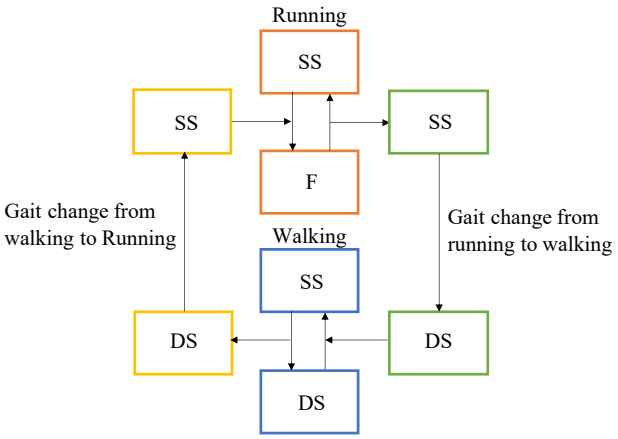

Fig. 2. Phase allocation in each gait

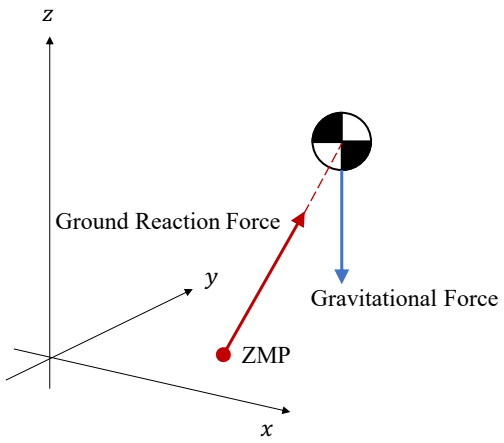

Fig. 3. The LIPM assumes that the ground reaction force does not produce the augular momentum around the CoM

planner. The stiffness of the LIPM is detemined by using the method described in Section III for steady-state walking, running, and transiton phases. Second, the position and velocity of the CoM and the position of the ZMP of the k-th contact phase is determined simultaneously by constraint solver for given footsteps. The CoM and ZMP trajectory are generated by interpolation of the postions determined above of each contact phase (Section II). The CoM is interpolated by closed-form solution of equation of its motion. Swing leg trajectory is generated by interpolation of the footprints. Torso trajectory is designed to satisfy condition about the position and velocity of the CoM and swing leg trajectory. Finally, desired joint angles and velocities are calculated by inverse kinematics for generated CoM, torso, swing leg trajectory.

B. Trajectory generation of CoM and ZMP

When the robot's motion consists of running and walking, there are three phases of contact situation. In the case of walking gait, the single stance (SS) phase and the double stance (DS) phase occur in turn. In the case of running gait, on the other hand, the contact state switches between the SS phase and the flight (F) phase. Gait transition is implemented within two phases consisting of the SS phase and the DS phase, as shown in Fig. 2 (the order depends on the direction of transition). In the case of the DS and SS phase, the following equation is derived from centroidal dynamics of the robot under the assumption that the angular momentum around the CoM is constant (see Fig.3).

$$\ddot{\boldsymbol{p}}(t) = \frac{1}{T(t)^2}(\boldsymbol{p}(t) - \boldsymbol{c}(t)) - \boldsymbol{g} \tag{1}$$

Here, $\boldsymbol{p}(t)$ is the CoM position, $\boldsymbol{c}(t)$ is the position of the ZMP, both are expressed in 3D coordinates. The time-varying parameter $T(t)$ determines the magnitude of the ground reaction force that acts on the CoM. In this paper, we call $T(t)$ the stiffness (more precisely, it is $1/T(t)^2$ which has the physical dimension of stiffness) This is an extension of the linear inverted pendulum mode (LIPM) [1] and we call it the time-varying LIPM. Let us consider a trajectory consisting of N contact phases. The value of the CoM position, velocity and the ZMP position at the beginning of the k-th phase ($1 \leq k \leq N$) is expressed as $\boldsymbol{p}_k, \boldsymbol{v}_k$ and $\boldsymbol{c}_k$. In our planning method, we assume that ZMP moves linearly during each phase (see Fig. 4).

$$\boldsymbol{c}(t) = \boldsymbol{c}_k + \frac{\boldsymbol{c}_{k+1} - \boldsymbol{c}_k}{\tau_k} t \quad (0 \leq t \leq \tau_k) \tag{2}$$

Here, τ_k is the duration of the k-th phase. In addition, we assume that $T(t)$ is piece-wise constant; that is, it takes a constant value during each contact phase. The value of T for each contact phase is determined by offline numerical optimization to realize appropriate vertical CoM motion for walking, running, and transitions between them (see Section

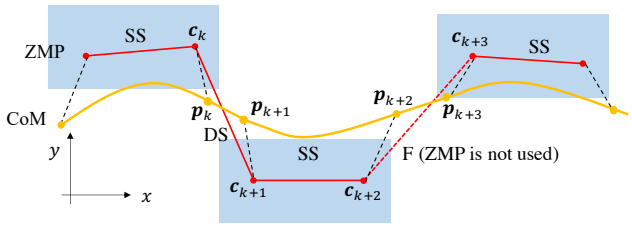

Fig. 4. Relationship of footprints, trajectories of the ZMP and the CoM

III). Under these assumptions, Eq. (1) can be solved analytically. Therefore, we can obtain the relationship between the position and the velocity of the CoM at the beginning of the k-th and $k+1$-th contact phases.

$$\boldsymbol{p}_{k+1} = \boldsymbol{g}T_k^2 + \boldsymbol{c}_{k+1} + (\boldsymbol{p}_k - \boldsymbol{c}_k - \boldsymbol{g}T_k^2)\cosh\left(\frac{\tau_k}{T_k}\right) + T_k\left(\boldsymbol{v}_k - \frac{\boldsymbol{c}_{k+1} - \boldsymbol{c}_k}{\tau_k}\right)\sinh\left(\frac{\tau_k}{T_k}\right) \tag{3}$$

$$\boldsymbol{v}_{k+1} = \frac{\boldsymbol{c}_{k+1} - \boldsymbol{c}_k}{\tau_k} + \frac{1}{T_k}(\boldsymbol{p}_k - \boldsymbol{c}_k - \boldsymbol{g}T_k^2)\sinh\left(\frac{\tau_k}{T_k}\right) + \left(\boldsymbol{v}_k - \frac{\boldsymbol{c}_{k+1} - \boldsymbol{c}_k}{\tau_k}\right)\cosh\left(\frac{\tau_k}{T_k}\right) \tag{4}$$

During the F phase, the CoM draws parabolic trajectory with its initial states equal to the terminal state of the former phase given by Eqs. (3) and (4). Thus, we obtain the relation of the CoM positions and velocities when the k-th phase is F as shown below.

$$\boldsymbol{p}_{k+1} = \boldsymbol{p}_k + \boldsymbol{v}_k\tau_k + \frac{1}{2}\boldsymbol{g}\tau_k^2 \tag{5}$$

$$\boldsymbol{v}_{k+1} = \boldsymbol{v}_k + \boldsymbol{g}\tau_k \tag{6}$$

Range constraints of the ZMP relative to support foot is formulated as below.

$$\underline{\boldsymbol{c}} \leq R_{\text{foot}}^T(\boldsymbol{c}_k - \boldsymbol{p}_{\text{foot}}) \leq \overline{\boldsymbol{c}} \tag{7}$$

Here, $\underline{\boldsymbol{c}}$ and $\overline{\boldsymbol{c}}$ are the bounds of the ZMP position relative to foot position $\boldsymbol{p}_{\text{foot}}$. R_{foot} represents the orientation of the foot. The inequalities are evaluated component-wise. To implement simultaneous planning of the CoM and the ZMP for given nominal CoM height, footsteps and the duration of each phase, we consider the following quadratic cost function to be minimized.

$$J = \frac{1}{2}\left[\sum_{k=1}^{N-1} f_k + \|\boldsymbol{p}_N - \boldsymbol{p}_{\text{f}}\|^2 + \|\boldsymbol{v}_N\|^2\right] \tag{8}$$

Here, f_k is the quadratic sum of the deviation from the constraints of Eqs. (3)-(7). The terminal costs are set to stop the CoM motion at the goal position $\boldsymbol{p}_{\text{f}}$. Simultaneous planning of the CoM and the ZMP is conducted by solving a problem to find $\boldsymbol{p}_k, \boldsymbol{v}_k, \boldsymbol{c}_k$ that minimize J. This problem is solved as a least squares problem by using Gauss-Newton method. The CoM trajectory is generated by interpolation among the computed CoM positions using the closed-form solution of equation of the CoM motion. This method is extended from our previously developed walking trajectory planner [16].

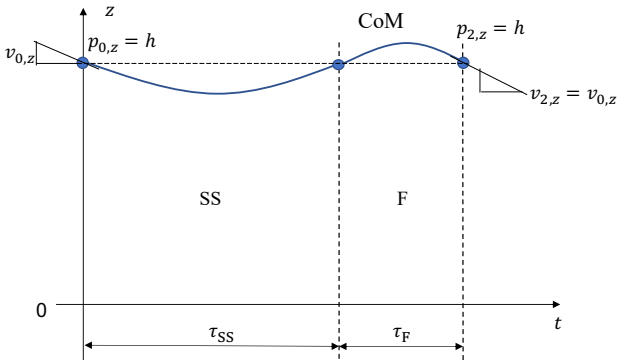

Fig. 5. CoM trajectory of running in the z direction that satisfy boundary conditions about its position and velocity

III. Computation of T_k for Walking, Running, and Transition between Them

A. Walking

Each gait requires a certain magnitude of vertical CoM movement to satisfy boundary conditions associated with it. To meet this requirement, the value of T is computed for each gait and transition phases. In the case of walking, let $\ddot{p}_z = 0, p_z = h$. Under this condition, T takes a unique value of $T_{\text{w}} = \sqrt{h/g_z}$ (c_z is always equal to zero).

B. Running

To generate vertical motion for running, we must find T which compensates for vertical deceleration during the flight phase. Consider a half cycle of running which contains one SS phase and one F phase. A problem of finding a combination of desired T and velocity of the CoM at the boundaries of this half cycle is formulated as a least squared optimization problem shown below.

$$\min_{T_{\text{r}}, v_{0,z}} \left\| \begin{bmatrix} h \\ v_{0,z} \end{bmatrix} - \begin{bmatrix} p_{2,z} \\ v_{2,z} \end{bmatrix} \right\|^2 \tag{9}$$

$$p_{2,z} = f_{\text{r}}(T_{\text{r}}, h, v_{0,z}, \tau_{\text{SS}}, \tau_{\text{F}}) \tag{10}$$

$$v_{2,z} = g_{\text{r}}(T_{\text{r}}, h, v_{0,z}, \tau_{\text{SS}}, \tau_{\text{F}}) \tag{11}$$

Here, $p_{2,z}, v_{0,z}, v_{2,z}$ are the position and the velocities of the CoM at the boundary of the half cycle. The right hand sides of Eqs. (10)-(11) are derived using Eqs. (3)-(6). Fig. 5 depicts the CoM trajectory in the z direction with the solution of Eqs. (9)-(11). One of the solutions, T_{r}, is used for the cyclic running phase. Another solution, $v_{0,z}$, is used as the parameter v_{r} to define the boundary condition of the least squares optimization problem for gait transition described in the next subsection.

C. Gait transition

Gait transition from walking to running completes within the DS phase and the SS phase. Because we determined

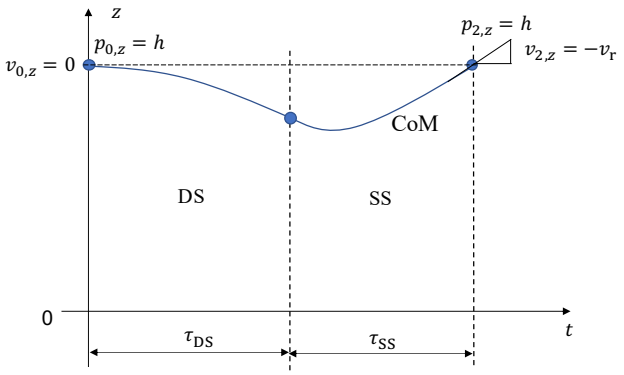

Fig. 6. CoM trajectory of transition from walking to running in the z direction that satisfy boundary conditions about its position and velocity

T_w so that the CoM moves at a constant height, the robot must start vertical motion from $v_{0,z} = 0$ to cyclic motion at the end of the SS phase with $v_{2,z} = -v_r$. A least-squares optimization problem is formulated to find T of each phase for desired vertical motion.

$$\min_{T_{t0},T_{t1}} \left\| \begin{bmatrix} h \\ -v_r \end{bmatrix} - \begin{bmatrix} p_{2,z} \\ v_{2,z} \end{bmatrix} \right\|^2 \tag{12}$$

$$p_{2,z} = f_t(T_{t0}, T_{t1}, h, \tau_{DS}, \tau_{SS}) \tag{13}$$

$$v_{2,z} = g_t(T_{t0}, T_{t1}, h, \tau_{DS}, \tau_{SS}) \tag{14}$$

The right hand sides of Eqs. (13)-(14) are derived using Eqs. (3)-(4). Eq. (12) is solved for given h, τ_{DS}, τ_{SS} in the same way as Eq. (9) is solved, and we obtain suitable T for each two phase as T_{t0}, T_{t1}. Fig. 6 illustrates the CoM trajectory in the z direction defined by the parameters obtained by solving Eqs. (12)-(14). We can reuse this result for gait change from running to walking because of the equations' symmetry. T_{t0} and T_{t1} are also used to start running and stop running immediately.

IV. Evaluation of Planning

A. Analysis of the computed T_k

The value of T_r with specified τ_{SS} and τ_F, varying from 0.10s to 0.39s with interval of 0.01s (total 900 combinations) is computed. The nominal height is fixed to $h = 0.95$[m] in the determination. The computation is performed in Python with the function `scipy.optimize.root` from the SciPy library. The result of the determination is plotted in Fig. 7. We can observe that proper solution is found in relatively broad range of combinations of contact phase duration, since the curved surface is smooth. The shorter the SS phase and the longer the F phase, the smaller T_r is computed. This result coincides with our intuition that larger ground reaction force is neccesary if the SS phase duration is relatively short compared to the F phase duration, because smaller T_r result in larger ground reaction force.

B. Analysis of plan with capture point

Capturability based analysis is conducted to evaluate a running trajectory generated by our method. We planned

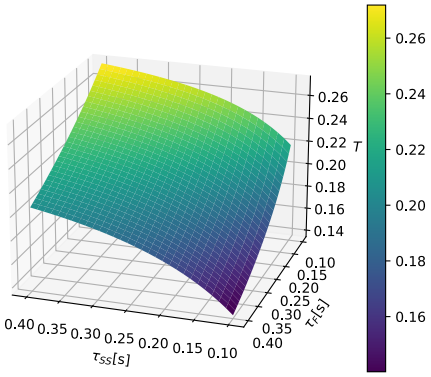

Fig. 7. The Distribution of T_r

TABLE I
Main parameters used in planning

parameter	value	T	value
h	0.95[m]	T_r	0.230
τ_{SS}	0.20[s]	T_{t0}	0.356
τ_{DS}	0.20[s]	T_{t1}	0.216
τ_F	0.20[s]		

a running trajectory consisting of 20 steps (8 steps acceleration/deceleration and 4 steps constant running) for evaluation. Values of some parameters used for planning are shown in Table I. Figure 8 shows the planned trajectory with max step length 0.50m. In the figure, the capture point is represented only during support phase. During the SS phase, capture point starts at the front half of the footprint, and there is a tendency that the initial position of the capture point approaches the edge of the footprint when the robot runs faster. In the acceleration, capture point starts near the center of the footprint compared to capture points in the deceleration. This result implies that it is possible to realize fast running compared to recently proposed LIPM based running trajectory generation guaranteeing landing capturability ([13]).

C. Comparison with SLIP

Comparison of CoM trajectory during the half cycle of periodic running generated by using the proposed method and the SLIP model is conducted. We selected a 3D-SLIP based method by Wensing and Orin [9] as a compared method. This method defines the half cycle of the CoM trajectory as a trajectory starts with when the CoM is at the top-of-flight (TOF) height, ends with when the CoM is at the next TOF height. The phase duration, contact angle, the CoM velocity in the y direction at the TOF, and the TOF height are optimized to generate a half cycle of the periodic CoM trajectory with the desired CoM velocity in the x direction at the TOF. Thus we first generated a half cycle of the SLIP model based trajectory for the TOF speed in the x direction of 3.5[m/s]. Using the optimized variables for the

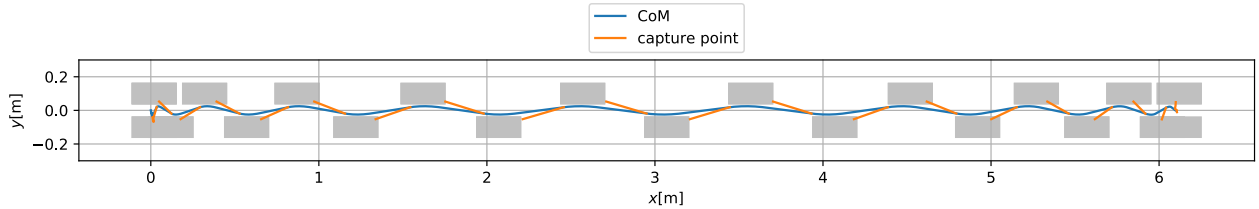

Fig. 8. 0-step capturability is almost broken when the robot runs fast

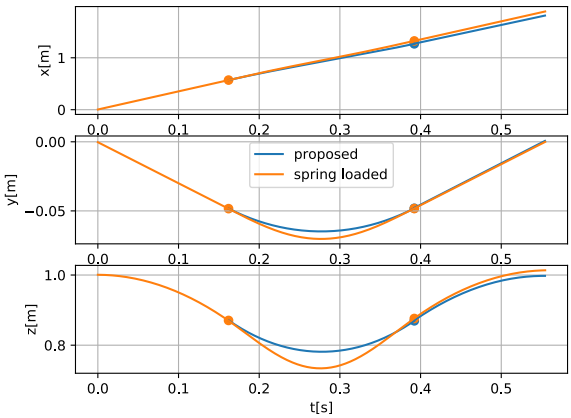

Fig. 9. Comparison of cyclic CoM trajectory

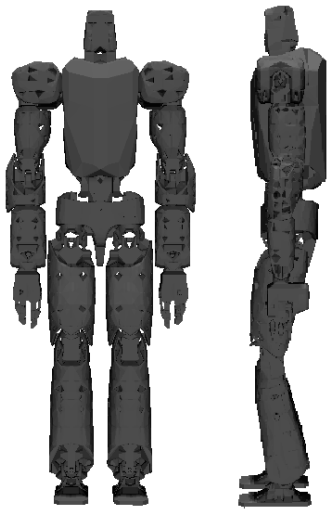

Fig. 10. Model of the humanoid robot Kaleido

generation of the SLIP model based trajectory, we generated a trajectory using Eqs. (3)-(6) as a trajectory based on our method. As a result, the two trajectories drew identical curve in the F phase. Figure 9 shows the result of the comparison in each direction. The difference of the two trajectories in the x and y direction are quite small, though our method uses closed-form solution while the SLIP-based method uses numerical integration. In addition, it takes 24.9s (with about 200 iterations) to compute parameters for the SLIP based trajectory, while it takes only 8ms (with 13 iterations) to compute T_r for the proposed method based trajectory (note that we implemented optimization using Python. Wensing and Orin[9] used MATLAB in their paper). The whole trajectory generation needs iteration of constraints solver, but at most 100 iterations (takes 1 to 2[s]) are practically sufficient. Therefore, our proposed method is computationally very efficient.

V. Simulation Experiments

A. Running

We chose a 32-DOF life-size humanoid robot Kaleido [18] (developed by Kawasaki Heavy Industries, Ltd.) as a simulation model (Fig. 10). We have found that tracking planned running trajectory requires sophisticated angular momentum control, which we have not yet implemented at the moment. To prevent the robot from falling without angular momentum control, we set moments of inertia of

TABLE II

Main parameters used in planning for simulation

parameter	value	T_k	value
h	0.95[m]	T_{w}	0.311
τ_{SS}	0.40[s]	T_{r}	0.259
τ_{DS}	0.15[s]	T_{t0}	0.363
τ_{F}	0.15[s]	T_{t1}	0.263

the base-link quite larger than the original value. We planned 20 steps trajectory (8 steps acceleration/deceleration and 4 steps constant running). Footstep positions is the same as the foot positions in the Fig. 8. Values of some input parameters used for planning are shown in Table II. In simulation, reference joint angles and velocities are calculated by inverse kinematics, and these are commanded to PD controller of each joint In addition, we incorporated controller to absorb landing impact of running ([17]). Figure 11 shows the reference and the simulation result of the CoM trajectory and the ground reaction force in the z direction. Vertical motion is generated in simulation as we expected especially during running in constant speed. Flight phase of about 0.10[s] is achieved, while planned flight phase length is 0.15[s]. During acceleration and deceleration, ground reaction force sometimes becomes 0 in the SS phase. This is because tracking performance of the CoM in the z direction is not good during the acceleration and deceleration, and result in

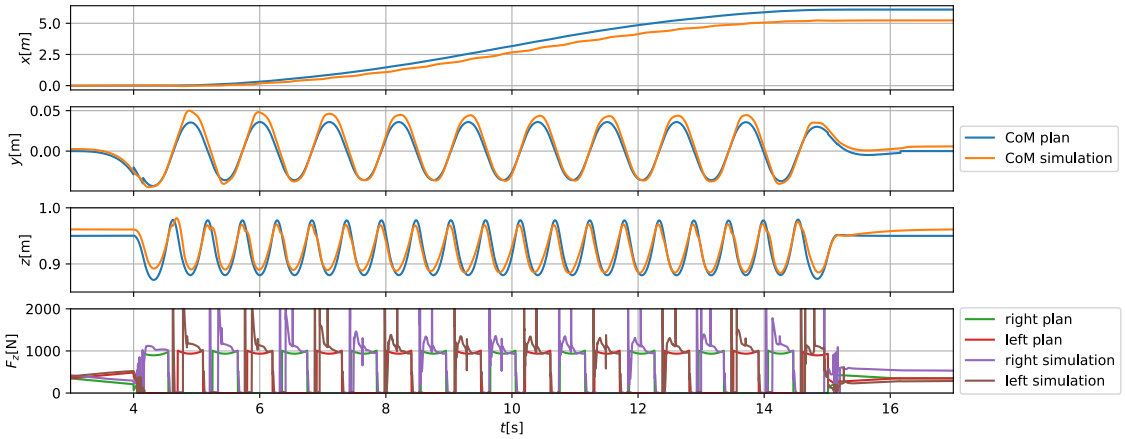

Fig. 11. CoM trajectory and ground reaction force in the z direction

the robot bouncing. This also seems to result in the error of the terminal position in the x direction because landing point of the robot moves backward when bouncing occurs.

B. Gait transition

We planned 20 steps trajectory that starts with walking (5 steps), transits to running (10 steps), and ends with walking (5 steps). The length of each step and main planning variable is the same as the running trajectory planning in the previous subsection. The robot could implement gait transition without falling down (see snapshots images in Fig. 12). Furthermore, the tracking performance of the CoM in the x direction is better than when whole trajectory consists of running. This seems to be the result of substituting walking for running during acceleration and deceleration, which indicates choosing suitable gait by gait transition may improve whole trajectory tracking performance. However, Fig. 13 shows the CoM trajectory in the z direction during walking does not track the reference. More precisely, the robot sinks during double stance and rises during single stance while walking. We must develop controller which can control motion of both walking and running. This issue remains to be our future work.

VI. Conclusion

In this paper, we proposed a trajectory generation method which enables gait transition using the time-varying LIPM. Comparison to conventional method indicated that our planner requires much less computation time to generate cyclic gait. Furthermore, several simulation results showed that our method is able to generate walking, running and gait transition trajectory. However, the tracking ability of the CoM trajectory to realize smooth gait transition in simulation needs to be improved. Developing a sophisticated trajectory tracking controller for both walking and running will be our future work.

References

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

[2] T. Koolen, T. de Boer, J. Rebula, A. Goswami, 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. 2012.

[3] J. Englsberger, C. Ott, A. Albu-Schäffer: “Three-dimensional bipedal walking control using Divergent Component of Motion”, IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013.

[4] S. Caron, Q. -C. Pham and Y. Nakamura: “ZMP Support Areas for Multicontact Mobility Under Frictional Constraints”, IEEE Transactions on Robotics, 2017.

[5] R. Blickhan, R. J. Full: “Similarity in multilegged locomotion: Bouncing like a monopode”, Journal of Comparative Physiology A, 1993.

[6] M. Ahmadi, M. Buehler: “Controlled Passive Dynamic Running Experiments With the ARL-Monopod II”, IEEE Transactions on Robotics and Automation, 2006.

[7] W. C. Martin, A. Wu, H. Geyer: “Experimental Evaluation of Deadbeat Running on the ATRIAS Biped”, IEEE Robotics and Automation Letters, 2017.

[8] W. L. Ma, S. Kolathaya, E. R. Ambrose, C. M. Hubicki, A. D. Ames: “Bipedal Robotic Running with DURUS-2D: Bridging the Gap between Theory and Experiment”, Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control, 2017.

[9] P. M. Wensing, D. E. Orin: “High-Speed Humanoid Running Through Control with a 3D-SLIP Model”, 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2013.

[10] R. Tajima, D. Honda, K. Suga: “Fast Running Experiments Involving a Humanoid Robot”, Proceedings of IEEE International Conference on Robotics and Automation, 2009.

[11] T. Takenaka, T. Matsumoto, T. Yoshiike, S. Shirokawa: “Real Time Motion Generation and Control for Biped Robot -2nd Report: Running Gait Pattern Generation-”, 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009.

[12] T. Kamioka, H. Kaneko, M. Kuroda, C. Tanaka, S. Shirokawa, M. Takeda, T. Yoshiike: “Dynamic Gait Transition between Walking, Running and Hopping for Push Recovery”, 2017 IEEE-RAS 17th International Conference on Humanoid Robotics, 2017.

[13] T. Sugihara K. Imanishi, T. Yamamoto, S. Caron: “3D biped locomotion control including seamless transition between walking and running via 3D ZMP manipulation”, IEEE International Conference on Robotics and Automation, 2021.

[14] Y. Liu, P. M. Wensing, D. E. Orin, Y. F. Zheng: “Dynamic Walking in a Humanoid Based on a 3D actuated Dual-SLIP Model”, 2015 IEEE/RSJ International Conference on Robotics and Automation, 2015.

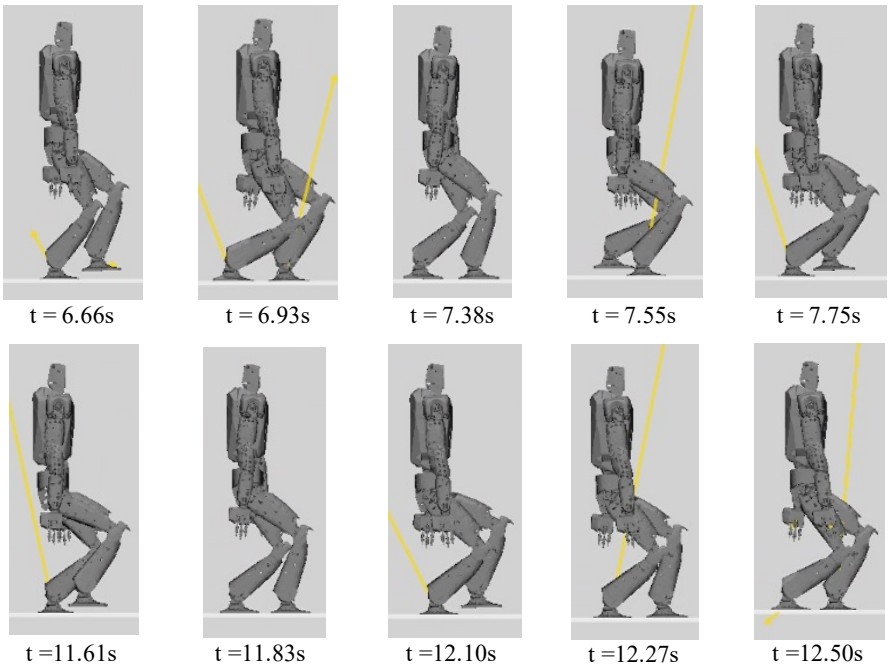

Fig. 12. Snapshots of the simulation including gait change. The Upper row represents the snapshots of transition from walking to running. The lower represents the snapshots of transition from running to walking. The Yellow line represents vector of the ground reaction force. Time stamps are consistent with the time scale of Fig. 13

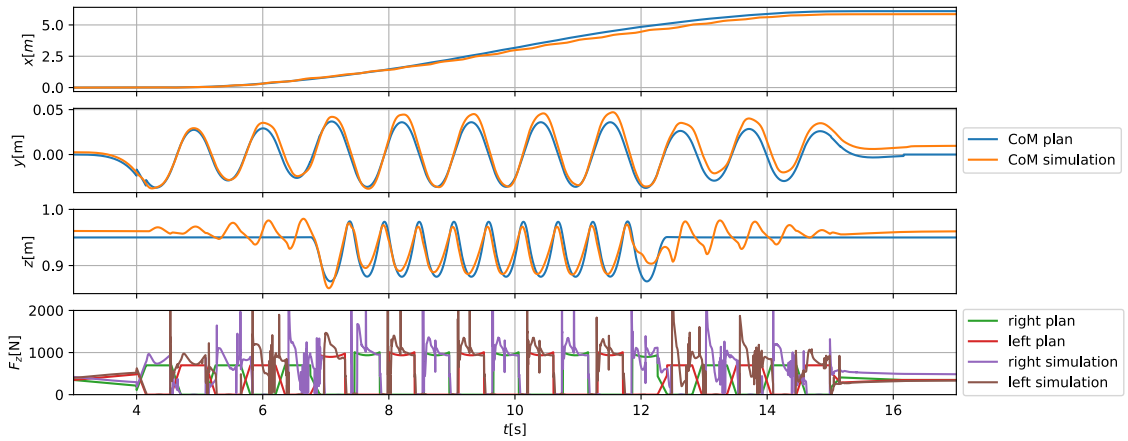

Fig. 13. The CoM trajectory and ground reaction force in the z direction including gait transition

[15] M. Shahbazi, R. Babuska, G. A. D. Lopes: "Unified Modeling and Control of Walking and Running on the Spring-Loaded Inverted Pendulum", IEEE Transactions on Robotics, 2016.
[16] K. Goto, Y. Tazaki, T. Suzuki: "Bipedal Locomotion Control Based on Simultaneous Trajectory and Foot Step Planning", Journal of Robotics and Mechatronics, Vol. 28, No. 4, pp. 533-542, 2016.
[17] S. Kajita, T. Nagasaki, K. Kaneko, K. Yokoi, K. Tanie: "A Running Controller of Humanoid Biped HRP-2LR", Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005.
[18] Y. Kakiuchi et al.: "Development of Life-sized Humanoid Robot Platform with Robustness fro Falling Down, Long Time Working and Error Occurrence" IEEE International Conference on Robotics and Automation, 2017.