Proceedings of 2004 IEEE/RSJ International Conference on
Intelligent Robots and Systems
September 28 - October 2, 2004, Sendai, Japan

A Running Experiment of Humanoid Biped

Takashi Nagasaki
Graduate School of
Systems and Information Engineering
University of Tsukuba
1-1-1 Tennodai, Tsukuba Ibaraki 305-0047
Japan
Email: t-nagasaki@aist.go.jp

Shuuji Kajita, Kenji Kaneko, Kazuhito Yokoi, Kazuo Tanie
Intelligent Systems Institute
National Institute of
Advanced Industrial Science and Technology (AIST)
Tsukuba Central 2, 1-1-1 Umezono, Tsukuba
Ibaraki 305-8568
Japan
Email: s.kajita@aist.go.jp

***Abstract*— Aiming for a humanoid robot of the next generation, we have been developing a biped which can jump and run. This paper introduces a method of running pattern generation and running experiment of biped robot HRP-2LR. Based on the physical parameters of HRP-2LR, running patterns are pre-calculated so that it follows the desired profiles of the total linear and angular momentum. For this purpose we used Resolved Momentum Control [16]. The vertical momentum is calculated considering the compliant elements in order to realize accurate flight duration. The horizontal momentum is calculated to satisfy the ZMP patterns given in advance. Using our pattern, HRP-2LR could successfully run with average speed of 0.16[m/s] with repeat flight phase 0.06 [s] and support phase 0.3 [s].**

I. Introduction

Research on humanoid robots is currently one of the most exciting topics in the field of robotics and there exist many projects [1]–[6]. Most of them focus on biped walking as an important subject and have already demonstrated reliable dynamic biped walking. Watching those successful demonstrations, one can ask a natural question, “Can we build a humanoid that can run?”

We believe this is worthwhile as a technical challenge for the following reasons. First, studying robot running will add new functions of mobility to humanoid robots. For example, jumping over large obstacles or a crevasse in the ground might be realized by a derivative of running control. Second, studying extreme situations will give us insights to improve the hardware itself. Current robots are too fragile to operate in any environment. Even when the robot operates at low speed, we must treat them carefully. We hope to overcome this fragility in the process of developing a running humanoid.

Running robots have been intensively studied by Raibert and his colleagues [7]. Their famous hopping robots driven by pneumatic and hydraulic actuators performed various actions including somersaults [8]. Using a similar control strategy, Hodgins simulated a running human in the computer graphics [9].

Ahmadi and Buehler studied running monopods from a standpoint of energy efficiency. Their ARL Monopod II [10] is an electrically powered running robot of 18 [kg] weight and could run at 1.25 [m/s] with a power expenditure of only 48 [W].

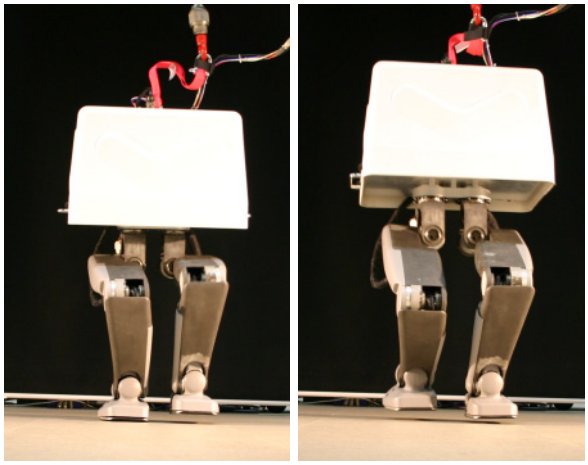

Fig. 1. Snapshots of running HRP-2LR

All of those robots have a spring mechanism to retrieve kinetic energy during running cycles. It is obvious that these springs help running but they might prevent ordinary humanoid activities such as walking, carrying objects and so on. Since our intention is to add a running function to a versatile humanoid robot, we started with a mechanism without springs. A similar approach is taken by Gienger et al. [1], Nagasaka et al. [11] and Chevallereau et al. [12].

In this paper, we report our first running experiment with humanoid biped HRP-2LR. In Section II, we explain the hardware of HRP-2LR. In Section III, a method for running pattern generation is described. In Section IV, the control system to stabilize the actual robot is outlined, then the experimental result is shown in Section V. We conclude this paper and address our future plans in Section VI.

II. Humanoid biped HRP-2LR

HRP-2LR was originally developed as an “Advanced Leg Module” in HRP [15]. To make it as light as possible, we removed onboard batteries and the dummy weight which emulated the mass of arms, head and chest. Through this remodeling, the total weight was reduced from 58.2 [kg] to 31.0 [kg] and the height shortened from 1.41 [m] to 1.27 [m]. Detailed specifications of HRP-2LR are shown in Tables I.

0-7803-8463-6/04/$20.00 ©2004 IEEE

TABLE I

SPECIFICATIONS OF HRP-2LR

Size	6D.O.F/Leg(Hip:3 Knee:1 Ankle:2)	
	Upper leg length:	300 [mm]
	Lower leg length:	300 [mm]
	Ankle-sole height:	93 [mm]
	Length between hip joints:	120 [mm]
	Toe-heel length:	170 [mm]
Weight	Legs:8.6 [kg/leg]×2 [legs] =	17.2 [kg]
	Controller:	7.0 [kg]
	Body structure	6.8 [kg]
	Total:	31.0 [kg]

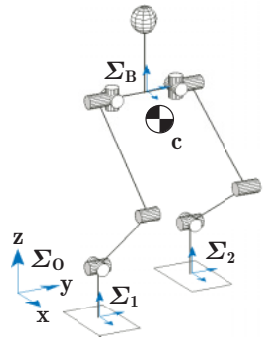

Fig. 2. *Structure of HRP-2LR*

The body contains a 3-axes acceleration sensor, three gyro sensors, twelve servo drivers and a CPU board (Pentium III, 933 [MHz]). Each foot is equipped with a 6-axes force sensor and rubber bushing which protects the sensor and robot from the touchdown impact.

III. RUNNING PATTERN GENERATION

A. Concept of pattern generation

To design running patterns, we plan the total(linear and angular) momentum for the robot and the foot velocities for each legs.

No matter how complex the robot's structure or behavior becomes, we can determine the position of the total center of mass (CoM) $\boldsymbol{c}(3\times1)$, the linear momentum $\mathcal{P}(3\times1)$ and the angular momentum $\mathcal{L}(3\times1)$ for the total mechanism. Dividing the total linear momentum $\mathcal{P}$ by the total mass of the robot m, we obtain the CoM velocity.

$$\frac{d}{dt}\boldsymbol{c} = \frac{\mathcal{P}}{m} \tag{1}$$

Thus, we can control the CoM position by manipulating the linear momentum. To calculate a pattern which gives the specified total momentum we use Resolved Momentum Control [16].

Fig.2 shows the structure of HRP-2LR with the base frame Σ_B on the waist link and the foot frames Σ_i ($i=1$ for the right and $i=2$ for the left). Let us introduce the velocities $(\boldsymbol{v}_B, \boldsymbol{\omega}_B)$ for Σ_B and the velocities $(\boldsymbol{v}_i, \boldsymbol{\omega}_i)$ for Σ_i. Hereafter, we put superscript d for the reference variables implying *designed* or *desired* values. When references of the linear momentum $\mathcal{P}^d$, the angular momentum $\mathcal{L}^d$ and the foot velocities $(\boldsymbol{v}_i^d, \boldsymbol{\omega}_i^d)$ are given, the corresponding waist velocities $(\boldsymbol{v}_B^d, \boldsymbol{\omega}_B^d)$ can be calculated as following,

$$\begin{bmatrix} \boldsymbol{v}_B^d \\ \boldsymbol{\omega}_B^d \end{bmatrix} = \begin{bmatrix} \boldsymbol{M}_B^* \\ \boldsymbol{H}_B^* \end{bmatrix}^{-1} \left(\begin{bmatrix} \mathcal{P}^d \\ \mathcal{L}^d \end{bmatrix} - \sum_{i=1}^{2} \begin{bmatrix} \boldsymbol{M}_i^* \\ \boldsymbol{H}_i^* \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_i^d \\ \boldsymbol{\omega}_i^d \end{bmatrix} \right). \tag{2}$$

$\boldsymbol{M}_B^*, \boldsymbol{H}_B^*, \boldsymbol{M}_i^*$ and $\boldsymbol{H}_i^*$ are modified inertia matrices, which can be calculated from physical parameters of the robot links and instantaneous configuration. The column vector $\dot{\boldsymbol{q}}_{leg_i}(6\times1)$ which contains joint velocities of each leg is calculated from $(\boldsymbol{v}_B^d, \boldsymbol{\omega}_B^d)$ and $(\boldsymbol{v}_i^d, \boldsymbol{\omega}_i^d)$ as follows,

$$\dot{\boldsymbol{q}}_{leg_i}^d = \boldsymbol{J}_{leg_i}^{-1} \begin{bmatrix} \boldsymbol{v}_i^d \\ \boldsymbol{\omega}_i^d \end{bmatrix} - \boldsymbol{J}_{leg_i}^{-1} \begin{bmatrix} \boldsymbol{E} & -\hat{\boldsymbol{r}}_{B\to i} \\ \boldsymbol{0} & \boldsymbol{E} \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_B^d \\ \boldsymbol{\omega}_B^d \end{bmatrix}, \tag{3}$$

where $\boldsymbol{J}_{leg_i}(6\times6)$ is the Jacobian matrix calculated from the leg configuration, $\boldsymbol{r}_{B\to i}(3\times1)$ is the position vector from the base link to the foot frame and $\hat{}$ is an operator which translates a vector of 3×1 into a skew symmetric matrix of 3×3 which is equivalent to a cross product. Using Eq.(2) and Eq.(3), we can calculate joint angular velocities to control the robot's CoM.

Following subsections review the method to calculate the reference linear momentum $\mathcal{P}^d$. The reference angular momentum $\mathcal{L}^d$ is given only the roll element as following,

$$\mathcal{L}_x^d = 0. \tag{4}$$

B. Momentum pattern

1) Vertical hopping pattern: When a robot is running, its CoM hops in a vertical direction. Let us define the duration of the flight phase as T_f and the duration of the support phase as T_s. Since the trajectories of the robot's CoM in the flight phase are limited to parabolas, planning of the support phases becomes like the following.

HRP-2LR is equipped with compliant elements in its feet. Moreover we cannot neglect servo compliance for high speed actions like hopping. When we control a robot without considering these effects, the flight duration tends to become longer than planned [13]. Therefore, we consider a hopping pattern generation which takes into account of the compliance effects through a spring-damper model(Fig.3).

In the support phase, to provide a reaction force pattern f_z^d, the position of the robot's CoM c_z and ankle joint a_z are described as follows.

$$m\ddot{c}_z = f_z^d - mg \tag{5}$$

$$-d\dot{a}_z - ka_z = f_z^d \tag{6}$$

The robot lifts off when the reaction force becomes zero, and the force should decrease smoothly toward the end of support phase(Fig.4).

$$f_z^d = \begin{cases} F_0 & 0 \le t \le \lambda T_s \\ F_0\{1 - (\frac{t-\lambda T_s}{(1-\lambda)T_s})^2\} & \lambda T_s \le t \le T_s \end{cases} \tag{7}$$

where F_0 is the constant value of vertical reaction force, λ is the parameter which determines the profile of f_z^d.

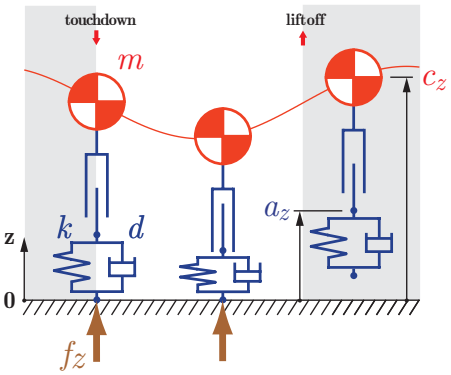

Fig. 3. *Inverted pendulum model with compliance element*

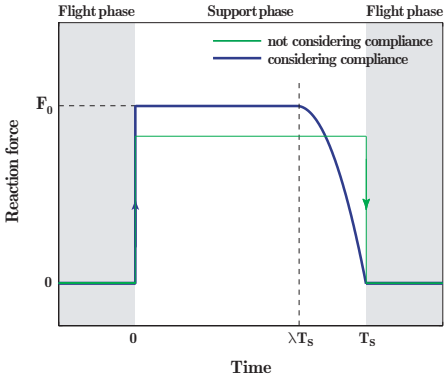

Fig. 4. *Reaction force pattern*

In the flight phase, the robot's CoM trajectory is a parabola, so the value at the touchdown (td) and liftoff (lo) of c_z must satisfy the following,

$$\begin{cases} \dot{c}_{z(lo)} - \dot{c}_{z(td)} = gT_f \\ c_{z(lo)} - c_{z(td)} = -\dot{c}_{z(lo)}T_f + \frac{1}{2}gT_f{}^2. \end{cases} \tag{8}$$

F_0 is calculated from Eq.(5), Eq.(7) and Eq.(8).

$$F_0 = \frac{3}{2+\lambda}(1 + \frac{T_f}{T_s})g \tag{9}$$

To limit the vertical motion of the robot's CoM in the support phase, λ should be close to 1.0. We choose $\lambda = 0.9$ by some experiments. Incidentally, when the robot is supported on both legs, it distributes the robot's weight to each legs appropriately according to a target ZMP pattern.

Using a pattern considering the foot compliance (the thick line of Fig.4) and a pattern not considering it (the thin line of Fig.4), we simulated a hopping with both of the legs. Fig.5 shows the resulted vertical forces. The flight duration becomes more accurate and the touchdown impact becomes smaller by considering the foot compliance.

2) Running phase pattern: Next, we plan the horizontal trajectory. We divide the whole motion among two sections,

- The running phase : The section in which the robot is running at a constant pace.

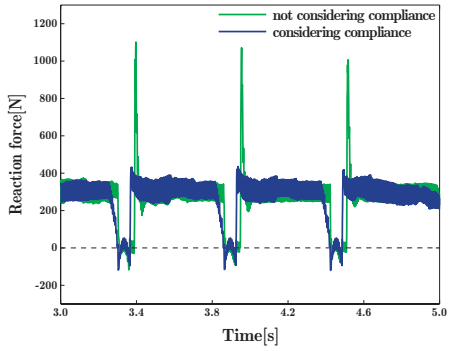

Fig. 5. *Reaction force at simulation*

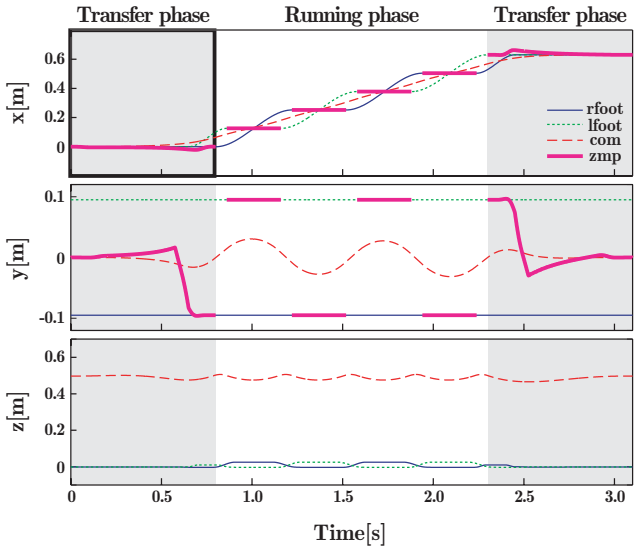

Fig. 6. *CoM and foots trajectories for running*

- The transfer phase : The section in which the robot accelerates and decelerates between standstill and the running phase.

and decide the running phase trajectory first(Fig.6).

The running phase consists of flight phases and support phases, and the condition which the robot's CoM must satisfy in the flight phase differs from the condition which the robot's CoM must satisfy in the support phase. In the flight phase, the speed of CoM is constant as shown in Eq.(10).

$$0 = \ddot{c}_x \tag{10}$$

In the support phase, it is necessary for the ZMP to lie within the limits of the support polygon. Therefore, to give the reference ZMP pattern p_x^d in advance, and using the vertical trajectory c_z^d already planned , we calculate the sagittal CoM trajectory from Eq.(11)

$$p_x^d = c_x - f(t)\ddot{c}_x \tag{11}$$
$$f(t) \equiv \frac{c_z^d}{\ddot{c_z}^d + g}$$

In order to solve these equations numerically, we treat them in discrete time domain with time step of Δt. By

representing $c_x(i\Delta t)$ as $c_{x(i)}$, $f(i\Delta t)$ as $f_{(i)}$ and $p_x^d(i\Delta t)$ as $p_{x(i)}^d$, Eq.(10) and Eq.(11) can be rewritten as

$$0 = \frac{1}{\Delta t^2} c_{x(i-1)} - \frac{2}{\Delta t^2} c_{x(i)} + \frac{1}{\Delta t^2} c_{x(i+1)}, \tag{12}$$

$$p_{x(i)}^d = -\frac{f_{(i)}}{\Delta t^2} c_{x(i-1)} + (1 + \frac{2f_{(i)}}{\Delta t^2}) c_{x(i)} - \frac{f_{(i)}}{\Delta t^2} c_{x(i+1)}. \tag{13}$$

In a flight phase($t = 0, \Delta t, \cdots, N_f \Delta t$), by arranging Eq.(12) in order of time, we get the following equation.

$$\overbrace{\begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}}^{\mathbf{0}} = \overbrace{\begin{bmatrix} \beta_0 & \alpha_0 & & \mathbf{0} \\ \alpha_1 & \beta_1 & \alpha_1 & \\ & \ddots & \ddots & \ddots \\ \mathbf{0} & & \alpha_{N_f} & \beta_{N_f} \end{bmatrix}}^{\boldsymbol{J}} \overbrace{\begin{pmatrix} c_{x(-1)} \\ c_{x(0)} \\ c_{x(1)} \\ \vdots \\ c_{x(N_f)} \\ c_{x(N_f+1)} \end{pmatrix}}^{\boldsymbol{C_f}} \tag{14}$$

$$\alpha_i \equiv \frac{1}{\Delta t^2}, \beta_i \equiv -\frac{2}{\Delta t^2}$$

In the same way, we arrange Eq.(13) in order of time for a support phase($t = 0, \Delta t, \cdots, N_s \Delta t$).

$$\overbrace{\begin{pmatrix} p_{x(0)}^d \\ p_{x(1)}^d \\ \vdots \\ p_{x(N_s)}^d \end{pmatrix}}^{\boldsymbol{P}} = \overbrace{\begin{bmatrix} \delta_0 & \gamma_0 & & \mathbf{0} \\ \gamma_1 & \delta_1 & \gamma_1 & \\ & \ddots & \ddots & \ddots \\ \mathbf{0} & & \gamma_{N_s} & \delta_{N_s} \end{bmatrix}}^{\boldsymbol{K}} \overbrace{\begin{pmatrix} c_{x(-1)} \\ c_{x(0)} \\ c_{x(1)} \\ \vdots \\ c_{x(N_s)} \\ c_{x(N_s+1)} \end{pmatrix}}^{\boldsymbol{C_s}} \tag{15}$$

$$\gamma_i \equiv -\frac{f_{(i)}}{\Delta t^2}, \delta_i \equiv 1 + \frac{2f_{(i)}}{\Delta t^2}$$

We combine Eq.(14) and Eq.(15) over the whole running phase, and get Eq.(16).

$$\begin{pmatrix} \mathbf{0'} \\ \boldsymbol{P_0} \\ \mathbf{0} \\ \boldsymbol{P_1} \\ \vdots \\ \mathbf{0'} \end{pmatrix} = \begin{bmatrix} \boldsymbol{J_0'} & & & & & \mathbf{0} \\ & \boldsymbol{K_0} & & & & \\ & & \boldsymbol{J_1} & & & \\ & & & \boldsymbol{K_1} & & \\ & & & & \ddots & \\ \mathbf{0} & & & & & \boldsymbol{J_m'} \end{bmatrix} \begin{pmatrix} \boldsymbol{C_{f0}'} \\ \boldsymbol{C_{s0}'} \\ \boldsymbol{C_{f1}'} \\ \boldsymbol{C_{s1}'} \\ \vdots \\ \boldsymbol{C_{fm}'} \end{pmatrix} \tag{16}$$

where the terms with $'$ indicate modifications for connecting.

To calculate the CoM trajectory c_x by solving Eq.(16), it is necessary to give the boundary conditions. We specify the speed at boundaries as v_f, thus

$$\begin{cases} \dot{c}_{x(0)} = v_f \\ \dot{c}_{x(N)} = v_f. \end{cases} \tag{17}$$

This makes the matrix of Eq.(16) to be tridiagonal, so we can solve it efficiently.

The ZMP pattern p_x^d should be chosen appropriately for the speed and the acceleration, we decide the relation of the ZMP pattern among the support phases as following,

$$p_{j+1} = p_j + (T_s + T_f) v_f, \tag{18}$$

where p_j is an element of $\boldsymbol{P_j}$ which is in Eq.(16).

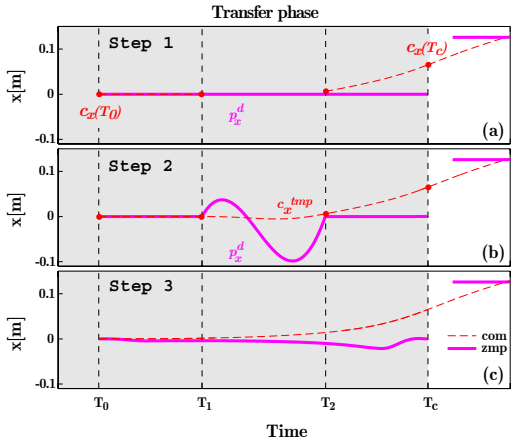

Fig. 7. *Planning transfer phase pattern (the entire graph corresponds to a thick line square on Fig.6)*

The CoM trajectory in the lateral plane is calculated in the same way for the sagittal plane. The boundary conditions which correspond to Eq.(17) are as following,

$$\begin{cases} c_{y(0)} = 0 \\ c_{y(N)} = 0. \end{cases} \tag{19}$$

3) Transfer phase pattern: In the transfer phase, the CoM trajectory is calculated in a different way from the running phase because the reference ZMP pattern is not obvious (Fig.7 : corresponding to thick lined square in Fig.6)

The initial state of the transfer phase is defined as $(c_x(T_0), \dot{c_x}(T_0)) = (0, 0)$ and the final state of the transfer phase is $(c_x(T_c), \dot{c_x}(T_c))$ which is defined by the running phase. Furthermore, it is necessary for the ZMP to lie within the limits of the support polygon each time. To satisfy these conditions, the transfer phase trajectory is planned by the following procedures.

Step1 We give an initial ZMP pattern p_x^d which lies within the limits of the support polygon (p_x^d is different from the final ZMP trajectory obtained in Step3). Next, T_1 and T_2 are set to satisfy $T_0 < T_1 < T_2 < T_c$, and we calculate a trajectory in $[T_0\ T_1]$ by integrating Eq.(11) from $(c_x(T_0), \dot{c_x}(T_0))$ forwards in time, and a trajectory in $[T_2\ T_c]$ by integrating from $(c_x(T_c), \dot{c_x}(T_c))$ backwards in time. (Fig.7(a))

Step2 The CoM trajectory in $[T_1\ T_2]$ is interpolated using a polynomial to keep continuity of acceleration. The CoM trajectory during the transfer phase is named c_x^{tmp} and the ZMP trajectory of c_x^{tmp} is named p_x^{tmp}. At this stage, the ZMP trajectory p_x^{tmp} comes off the range of the support polygon in some cases. (Fig.7(b))

Step3 To satisfy that the ZMP trajectory always lies within the support polygon, the ZMP trajectory p_x^{tmp} is brought close to p_x^d by modification of the CoM trajectory c_x^{tmp}. Using $\boldsymbol{K}$ in Eq.(15), the correcting function Δc_x which makes p_x^{tmp}

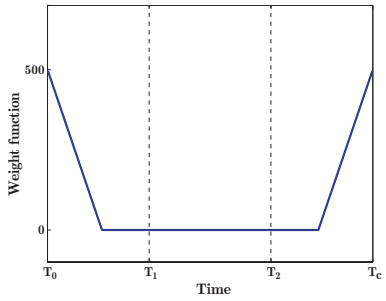

Fig. 8. *Weight function*

coincide with p_x^d is as follows [14],

$$\Delta c_x = \boldsymbol{K}^{-1}(p_x^d - p_x^{tmp}) \tag{20}$$

However, both ends of the CoM trajectory c_x^{tmp} should not be modified because the both ends are already fixed to $(c_x(T_0), \dot{c_x}(T_0))$ and $(c_x(T_c), \dot{c_x}(T_c))$ respectively. By introducing a weight function $\boldsymbol{W}$, we can limit values of Δc_x.

$$\Delta c_x = \begin{bmatrix} \boldsymbol{K} \\ \boldsymbol{W} \end{bmatrix}^{\dagger} \begin{bmatrix} p_x^d - p_x^{tmp} \\ \boldsymbol{0} \end{bmatrix} \tag{21}$$

where † denotes the pseudo-inverse of the respective matrix. The weight function $\boldsymbol{W}$ is a diagonal matrix and the diagonal elements are shown in Fig.8.
The modified CoM trajectory by using c_x^{tmp} is shown Fig.7(c).

IV. Running Controller

Fig.9 illustrates the entire running control system including pattern generations. The designed total momentum $\mathcal{P}^d, \mathcal{L}^d$ and the foot motion $\boldsymbol{v}_i^d, \boldsymbol{\omega}_i^d$ $(i = 1, 2)$ are processed by Resolved Momentum Control and we obtain a running pattern consists of the following parameters.

$\boldsymbol{q}^d$	Joint angles $\boldsymbol{q}^d \equiv [\boldsymbol{q}_{leg_1}^T \boldsymbol{q}_{leg_2}^T]^T$
ϕ^d, θ^d, ψ^d	Body posture ($\boldsymbol{R}_B^d$ in roll-pitch-yaw)
$\boldsymbol{\omega}_B^d$	Body angular velocity
$\boldsymbol{f}_i^d, \boldsymbol{\tau}_i^d$	Foot force and moment
phase	Running phase (support leg) (Right,Left,Double,Flight)

These data are fed to *running controller* where the target joint angles $\boldsymbol{q}^d$ are modified so that the robot can continue running under disturbances. For this purpose, the running controller uses foot force sensor information and actual body posture estimated by Kalman filter handling gyro and acceleration sensors.

Fig.10 depicts the inside architecture of the running controller. Since HRP-2LR cannot measure absolute body position, its reference $\boldsymbol{p}_B^d$ is useless (see trash-can in Fig.9). Instead of this, the controller adopts a local coordinate system where the center of the body is the origin. All information are casted into the local Cartesian coordinate using forward kinematics, and *stabilizing algorithms* works in this domain. For example, when the robot suffers excessive vertical floor force at landing, it shortens the both of the legs[1]. Finally, the results of modifications are transformed into joint angles using inverse kinematics and send to the local PD servo controller of HRP-2LR.

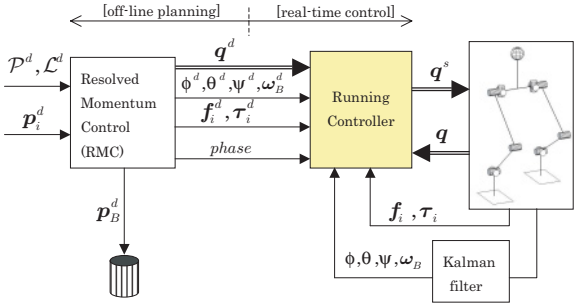

Fig. 9. Outline of HRP-2LR control system

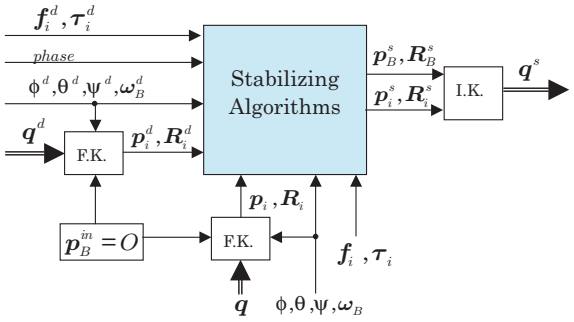

Fig. 10. Architecture of the running controller. F.K.:Forward Kinematics, I.K.:Inverse Kinematics

V. Running Experiment

Fig.11 shows a running experiment of HRP-2LR. The parameters of the running pattern are listed in Table II.

Using a pre-calculated running pattern and the running controller, HRP-2LR could successfully run with an average speed of 0.16 [m/s]. This speed was 64% of planned value (0.25 [m/s]), due to slips between the robot's sole and the ground. The running cycle, however, was controlled as planned. We can confirm this by watching the vertical forces measured at the robot's feet (Fig. 12).

VI. Conclusions

In this paper, we introduced a method of running pattern generation and experiment of biped robot HRP-2LR. Based on the physical parameters of HRP-2LR, running patterns were pre-calculated so that it follows the desired profiles of the total linear and angular momentum. For this purpose we used Resolved Momentum Control [16]. The vertical momentum was decided to consider the compliant

[1]More comprehensive stabilizing algorithms will appear in our next report.

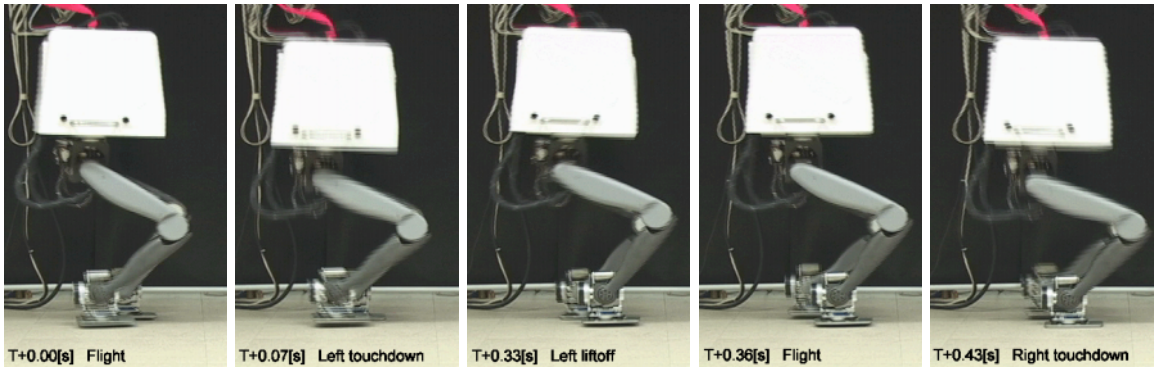

Fig. 11. Running experiment of HRP-2LR. The robot is running from left to right with average speed of 0.16 [m/s].

TABLE II

RUNNING PATTERN AND RESULT

Planned	Support time T_s:	0.3 [s]
	Flight time T_f:	0.06 [s]
	Travel distance:	0.81 [m]
	Average speed:	0.25 [m/s]
	Maximum foot height:	0.025 [m]
	Steps:	9 [step]
Result	Traveled distance:	0.55 [m]
	Average speed:	0.16 [m/s]

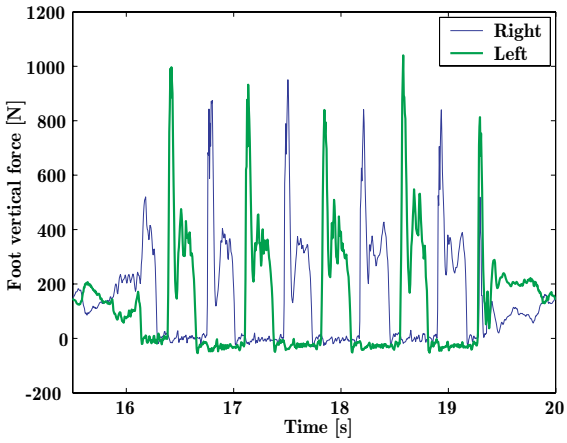

Fig. 12. Vertical foot force at running experiment

elements in order to realize accurate flight duration, and the horizontal momentum was calculated to satisfy the ZMP patterns given in advance. Using our pattern, HRP-2LR could successfully run with average speed of 0.16[m/s] with repeat flight phase 0.06 [s] and support phase 0.3 [s].

In the experiments of this paper, we deliberately used short flight time and slow speed due to joint velocity restrictions. The realization of faster running will be our next target.

ACKNOWLEDGMENT

We thank people of Kawada Industries, Inc. especially Jiro Sakurai, Toshikazu Kawasaki and Takakatsu Isozumi for their technical support. We also thank Hirohisa Hirukawa, Fumio Kanehiro, Kiyoshi Fujiwara, Kensuke Harada and Hajime Saito of Humanoid Research Group, AIST for their excellent advice.

REFERENCES

[1] Gienger, M., et.al, "Toward the Design of a Biped Jogging Robot," Proc. of the 2001 ICRA, pp.4140–4145, 2001.
[2] Hirai, K., Hirose, M., Haikawa, Y. and Takenaka, T., "The Development of Honda Humanoid Robot," Proc. of the 1998 ICRA, pp.1321–1326, 1998.
[3] Inoue, H., Tachi, S., Nakamura, Y., Hirai, K., et.al, "Overview of Humanoid Robotics Project of METI," Proc. Int. Symp. Robotics, pp.1478–1482, 2001.
[4] Nishiwaki, K., Sugihara, T., Kagami, S., Kanehiro, F., Inaba, M., and Inoue, H., "Design and Development of Research Platform for Perception-Action Integration in Humanoid Robot: H6," Proc. Int. Conference on Intelligent Robots and Systems, pp.1559–1564, 2000.
[5] Yamaguchi, J., Soga, E., Inoue, S. and Takanishi, A., "Development of a Bipedal Humanoid Robot – Control Method of Whole Body Cooperative Dynamic Biped Walking –," Proc. of the 1999 ICRA, pp.368–374, 1999.
[6] Kim, J., Oh, J., "Walking Control of the Humanoid Platform KHR-1 based on Torque Feedback Control," Proc. of the 2004 ICRA, pp.623–628, 2004.
[7] Raibert, M., *Legged Robots that Balance*, Cambridge, MA, MIT Press, 1986.
[8] Playter, Robert R. and Raibert, Marc H., "Control of a Biped Somersault in 3D," Proc. of IFToMM-jc International Symposium on Theory of Machines and Mechanisms (in Nagoya, Japan), pp.669–674, 1992.
[9] Hodgins, J. K.,"Three-Dimensional Human Running," Proc. of the 1996 ICRA, pp.3271-3277, 1996.
[10] Ahmadi, M. and Buehler, M., "The ARL Monopod II Running Robot: Control and Energetics," Proc. of the 1999 ICRA, pp.1689-1694, 1999.
[11] Nagasaka, K., Kuroki, Y., Suzuki, S., Itoh, Y., Yamaguchi, J., "Integrated Motion Control for Walking, Jumping and Running on a Small Bipedal Entertainment Robot," Proc. of the 2004 ICRA, pp.3189–3194, 2004.
[12] C. Chevallereau, E.R. Westervelt, and J.W. Grizzle, Asymptotically Stable Running for a Five-Link, Four-Actuator, Planar Bipedal Robot, Pre-print, paper submitted to IJRR. Paper, 2004.
[13] Kajita, S. et al.,"A Hop towards Running Humanoid Biped," Proc. of the 2004 ICRA, pp.629–635, 2004.
[14] Kagami, S., Nishiwaki, K., Kitagawa, T., Sugihara, T., Inaba, M. and Inoue, H., "A Fast Generation Method of a Dynamically Stable Humanoid Robot Trajectory with Enhanced ZMP Constraint," Proc. of Int. Conference on Humanoid Robotics, 2000.
[15] Kaneko, K. et al.,"Design of Advanced Leg Module for Humanoid Robotics Project of METI," Proc. of the 2002 ICRA, pp.38–45, 2002.
[16] Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K. and Hirukawa, H., "Resolved Momentum Control: Humanoid Motion Planning based on the Linear and Angular Momentum," Proc. of the 2003 IROS, pp.1644–1650, 2003.