Detection and Stabilization of Hybrid Periodic Orbits of Passive Running Robots

Sang-Ho Hyon Takashi Emura
Department of Bioengineering and Robotics
Tohoku University
Sendai 980-8579, Japan
Email: sangho@ieee.org

Tetsushi Ueta
Center for Advanced Information Technology
Tokushima University
Tokushima 770-8506, Japan
Email: tetsushi@is.tokushima-u.ac.jp

***Abstract*— In this paper a new algorithm to detect hybrid periodic orbits of autonomous hybrid dynamical system is developed. Conventional Newton algorithm is modified so that it suits to the analysis of Poincaré return map of hybrid dynamical systems that include multiple phases (modes) and discrete jumps. Then, the algorithm is applied to a specific example; planar one-legged robot model having a springy leg and a compliant hip joint. With the algorithm, passive running gaits of the one-legged robot are automatically detected for various parameter sets and initial conditions. The analysis of the characteristic multiplier of the return map revealed the stability and the bifurcation of the passive running gaits. Two kinds of controllers that achieve orbital stabilization are presented. A similarity is found between the detection algorithm and the stabilizing controller. The algorithm can be applied to any kinds of the robots (e.g. walking robot).**

I. Introduction

After the Raibert's excellent works [1], theoretical aspects of basic one-legged running model has been deeply studied by Koditschek and his coworkers [2] [3] [4]. In the fast running control, energy-efficiency is especially crucial for autonomous robots (including biped humanoid robots or quadruped robots) because it directly extends their operation time. In this connection, there are some remarkable researches on passive running, where the passive running means "unforced" periodic running. Tompson and Raibert showed that spring-driven one-legged hopping robot can hop without any inputs, provided the initial conditions were appropriately chosen [5]. Ahmadi and Buehler applied Raibert's algorithm to this robot and realized energy-efficient hopping [6]. François and Samson derived a rather systematic controller based on linearization of the periodic orbit [7].

Our goal is to study the mechanism of energy-efficient legged locomotion and to realize it in legged robots. To this end, a general approach is established:

1) Explore the "passive dynamics" of the model (study the periodic orbits if they exit),
2) Design a "main controller" that makes the system trajectory to be *reccurent* (prevent the robots from falling),
3) Design "sub controllers" that adaptively stabilize the recurrent trajectories to optimal periodic orbits (passive orbits if they exist),

where "optimal" periodic orbits means passive (zero input) periodic orbits for passively driven legged robots.

However, it is very difficult to obtain analytical solution even for simple 2 degrees of freedom (DOF) case because the system is not integrable [3]. Therefore, numerical method must be employed. If we can "luckily" find passive (unforced) periodic gaits, the next stage is their stability analysis. When the system has some controllability, we can design controllers, which asymptotically stabilize the gaits, even if passive gaits are unstable. This is a stabilization problem of nonlinear hybrid oscillator, which has not been well discussed. Thus obtained limit cycle is utmost important for energy-efficient locomotion because the control input becomes zero when the solution lies on the limit cycle. Moreover, if the controller has large *region of attraction*, it will be a good template controller for more complex robot models.

In this paper, we develop a new simple algorithm to detect hybrid periodic orbits of autonomous hybrid dynamical system and apply it to one-legged robot [8]. In Section II, the new detection algorithm is derived according to the previous work [9]. Conventional Newton algorithm for continuous dynamical system (e.g. [10]) is modified so that it suits to the analysis of the *Poincaré return map* (simply called "return map") of the hybrid system that includes multiple phases (modes). Using this algorithm, search space is reduced from $2n$ to $2n-1$ or less. This leads to a great help for the analysis of the full-dimensional hybrid system behavior. Also, a modification is made so that the algorithm can deal with discrete jumps, which naturally participate in the dynamics of legged robots. With this algorithm, the stability of hybrid periodic orbits also can be exactly identified (we mean "exactly" in numerical sense).

The algorithm derived in Section II is applied to a planar one-legged robot model in Section III. We will show the passive running gaits and their stability in terms of return map.

Based on the stability analysis, Section IV shows controllers that achieve *orbital stability*. First, a local linear feedback controller is derived. Then, we will introduce our previously derived *Delayed-Feedback Controller*, combined with *Energy-Preserving Controller* [12]. The controllers are derived along with our general approach 2) and 3) listed above. They actually asymptotically stabilize unknown (multi) periodic gaits, and additional adaptive energy controllers achieves stable passive running. Then, a close relationship between the detection algorithm and the Delayed-Feedback Controller is discussed.

II. Detection of hybrid periodic orbits

To search passive running gaits, we construct return map with the control input being zero, because the fixed points of the map mean the existence of passive running gaits. Newton-Raphson method is widely used for finding the solution of algebraic equation. However, running robot is a quite complex hybrid system composed of stance phase, flight phase and other discrete events. Therefore, Newton-Raphson method should be modified appropriately.

A. Autonomous hybrid dynamical system

The locomotion dynamics of n-DOF legged model can be represented by an autonomous (s-modal) hybrid dynamical system composed of differential equations and algebraic equations:

- Continuous transition:

$$\dot{\underline{x}}_i = f_i(\underline{x}_i) \quad (i = 1, 2, \cdots, s) \tag{1}$$

- Discrete jump:

$$\underline{x}_{i+} = h_i(\underline{x}_{i-}) \quad (i = 1, 2, \cdots, s) \tag{2}$$

- Events:

$$\Pi_i := \{\underline{x}_i | S_i(\underline{x}_i) = 0\} \quad (i = 1, 2, \cdots, s) \tag{3}$$

Here, $\underline{x_i} \in R^{2n}$ is the state vector, i and s are the index and the number of the phases (modes) respectively. $f_i \in R^{2n}$ is the continuous vector fields, and $h_i \in R^{2n}$ is the jump equation, by which the states reset to the initial position of each phase. The variables $\underline{x}_{i+}$ and $\underline{x}_{i-}$ mean the states just after and before events respectively.

The equation of motion of continuous transition can be represented by autonomous dynamical systems because there are no control inputs, nor specific “clock time”. Note that DOF of each phase could be different; it could be n or less. Therefore, some of the elements in the state vector $\underline{x}_i$ do not participate in the dynamics in some phases. In this case, they become dependent variables, and the update of these variables should be calculated by kinematic constraint equations subject to the corresponding phases.

On the other hand, the equations of motion at discrete events can be represented by algebraic equations. They mainly contribute to describe the behavior of the impact phenomena, which occurs when the leg strikes the ground (the other is to describe “reset” of the position at the end of one stride). This consideration complicates the analysis, but they make the model more realistic.

The solution of this hybrid dynamical system evolves as the following manner (see Fig. 1) : $\underline{x}_i(t)$ starts from the initial condition $\underline{x}_i(0) = \underline{x}_{i+}$ and evolves along to Eq. (1). When $\underline{x}_i(t)$ reaches the hyper-surface $\Pi_{(i+1)}$, it jumps according to Eq. (2), then it evolves in the next phase.

B. The algorithm

The purpose of this section is to provide the searching algorithm of the hybrid periodic orbit, which is indicated by γ in Fig. 1. Let's begin with the single modal system, that is, $s = 1$ in Eq. (1) – (3) (We omit the mode-indicating subscript 1 for simplicity).

By choosing Π as the local *cross sections* [1] , the return map can be represented by:

$$\underline{x}(k+1) = P(\underline{x}(k)), \tag{4}$$

where k means the step of running. With this setup, what we have to do is; finding the solutions of:

$$G(\underline{x}) := \underline{x} - P(\underline{x}) = 0. \tag{5}$$

To this end, we apply the conventional Newton-Raphson method. The variation of P corresponding to a bit change of $\underline{x}$ can be obtained from Taylor expansion:

$$P(\underline{x} + \Delta\underline{x}) = P(\underline{x}) + DP(\underline{x})\Delta\underline{x} + O(\underline{x}), \tag{6}$$

where $DP(\underline{x}) = \frac{\partial P(\underline{x})}{\partial \underline{x}}$ and $O(\underline{x})$ is the higher-order term. Using (5), we have:

$$\begin{aligned} &DG(\underline{x})\Delta\underline{x} = -G(\underline{x}) \\ &\Rightarrow \Delta\underline{x} = (I - DP(\underline{x}))^{-1}(P(\underline{x}) - \underline{x}), \end{aligned} \tag{7}$$

Therefore, for a given initial guess $\underline{x}^0$, the searching algorithm can be written as:

$$\underline{x}^{j+1} = \underline{x}^j + (I - DP(\underline{x}^j))^{-1}(P(\underline{x}^j) - \underline{x}^j), \tag{8}$$

where the superscript j is the iteration number of the Newton algorithm.

In this algorithm, the update of the solution should be executed on Π, therefore DP must be calculated by:

$$DP(\underline{x}) = \left[I - \frac{1}{\frac{\partial S}{\partial \underline{x}} f(\underline{x})} f(\underline{x}) \frac{\partial S}{\partial \underline{x}} \right] \Phi_T(\underline{x}), \tag{9}$$

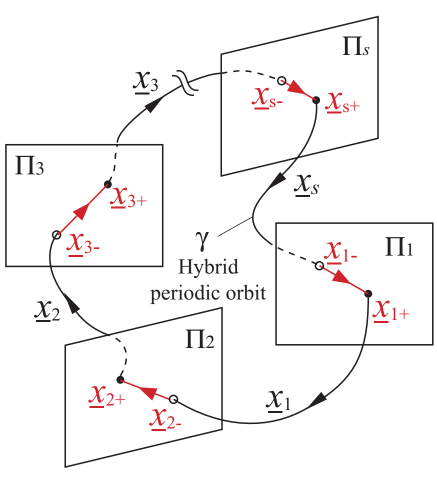

Fig. 1. Autonomous hybrid system and its hybrid periodic solution γ. $\underline{x}_i$ are the states of each phase (modes), and Π_i are Poincaré' cross sections.

[1]The cross section Π must be chosen to meet the *transversality condition*: $\frac{DS}{D\underline{x}} f(\underline{x}) \neq 0$. See [13].

where $\Phi_T(\underline{x})$ is the *principal matrix solution* [14], the solution $X(t)$ of the following *variational equation* evaluated at time $t = T$ (T is the period; the solution starts from the section at time 0 and returns after time T):

$$\frac{dX}{dt} = Df(\underline{x})X \tag{10}$$

$$\text{with} \quad X(0) = \frac{\partial \phi_t(x_0)}{\partial x_0} = I, \tag{11}$$

where ϕ_t is the flow of the vector field $f(\underline{x})$. Note that in autonomous hybrid system the period T depends on initial conditions and automatically updated as the algorithm proceeds. Usually, the principal matrix solution $\Phi_t(\underline{x})$ are obtained numerically in parallel with $\underline{x(t)}$. For example, constructing "extended state vector" composed of original states and the elements of identity matrix and evolve it at once in the ODE solver.

In the case of continuous system, the above equations are sufficient to find periodic orbits. If there is a jump in the solution $\underline{x}$, however, DP should be processed appropriately as follows. Suppose the discrete jump is represented by an algebraic equation:

$$\underline{x}_+ = h(\underline{x}_-). \tag{12}$$

Then, from the chain rule of differentiation, Φ changes according to:

$$\Phi_+ = \frac{\partial h}{\partial \underline{x}_-}\Phi_-, \tag{13}$$

where Φ_- and Φ_+ represents the value of Φ evaluated just before and after jump respectively.

Moreover, since we have defied the return map on the local cross section in full-dimensional space, thus obtained principal matrix solution always includes *characteristic multiplier* (eigen value of DP) 1, whose associated eigen vector coincide with the vector field f [14]. Since this "trivial" multiplier lead to a difficulty in further bifurcation analysis, Φ must be reduced by:

$$\Phi_{red} = \frac{\partial \eta}{\partial \underline{x}} \cdot \Phi \cdot \frac{\partial \eta^{-1}}{\partial \underline{y}}, \tag{14}$$

where $\underline{y}$ is the local coordinates, the explicit expressions of $S(\underline{x}) = 0$, and $\eta : \underline{x} \mapsto \underline{y}$ is the *projection* of $2n$-dimensional manifold to $(2n-1)$-dimensional manifold. Its inverse η^{-1} is *embedding* to $2n$-dimensional manifold [15]. The examples of local coordinates are given in the next section.

Finally, combining (9), (13) and (14) results in:

$$DP(\underline{x}) = \frac{\partial \eta}{\partial \underline{x}} \cdot \frac{\partial h}{\partial \underline{x}_-} \cdot \left[I - \frac{1}{\frac{\partial S}{\partial \underline{x}} f(\underline{x})} f(\underline{x}) \frac{\partial S}{\partial \underline{x}} \right] \cdot \Phi_T(\underline{x}) \cdot \frac{\partial \eta^{-1}}{\partial \underline{y}}, \tag{15}$$

which is used in the update law (8). This formula is much simpler than Hisken's one (compare Eq. (15) with Eq. (57)–(60) in [11]), which focuses on the general representation of hybrid dynamical system. We need only the algorithm that detects the periodic orbits of "autonomous" hybrid dynamical system, where the jump is always autonomous. For the details of (9) and (14), see [9].

Now, it is very easy to consider the multiple modal case, that is, when $s \geq 2$ in Eq. (1) – (3). In this case, we only have to take the following procedure.

(1) Take the jump solution $\underline{x}_{i+}$ as the initial condition of the current phase $\underline{x}_i(0)$.
(2) Restore Φ_{iT_i} into the memory and take identity matrix I as the initial conditions for the variational equation of each phase.
(3) At the Poincaré section Π_1, combine Φ_is by:

$$\Phi_T(\underline{x}) = \Phi_{1T_1}(\underline{x}_1) \cdot \Phi_{2T_2}(\underline{x}_2) \cdots \Phi_{sT_s}(\underline{x}_s), \tag{16}$$

(4) Apply Eq. (15) to obtain $DP(\underline{x})$.

Note that Eq. (16) comes from the fact:

$$P = P_1 \circ P_2 \circ \cdots \circ P_s, \tag{17}$$

where P_i is the return maps that brings $\underline{x_i}(0)$ to $\underline{x}_{(i+1)-}$, and from the chain rule. Note that the period of the orbit is $T = T_1 + T_2 + \cdots + T_s$.

C. Stability analysis

Once periodic solutions are obtained using the algorithm, we can easily check their stability by the following theorem (see e.g. [14]).

Theorem 2.1: If all the eigenvalues of $DP(\underline{x})$ have modulus less than 1, then the closed orbit γ is asymptotically *orbitally stable*. If one of the eigenvalues of $DP(\underline{x})$ have modulus greater than 1, then the closed orbit γ is asymptotically *orbitally unstable*.

III. Application to a one-legged robot

A. Model description

Figure 2 shows the model of the planar one-legged running robot considered here. The robot is attached not only with a leg spring but also a hip spring.

We impose the following assumptions on this model.

(A) The center of mass (C.M.) of the body leis on the longitudinal axis of the leg (e.g. on hip joint in Fig. 2)
(B) Mass of the foot (unsprung mass) is negligible
(C) The foot does not bounce back, nor slip the ground (*inelastic impulse* assumption)
(D) The springs are loss-less

The equations of motion are composed of two phases; stance and flight phase, triggered by two discrete events; lift-off and touchdown. We use event-indicating subscripts for variables. For example, $\dot{x}_{lo}$ is the forward velocity of C.M. at lift-off, $\dot{\theta}_{td+}$ is the angular velocity of the leg just before touchdown, θ_{td} is the leg angle of just before, or, just after touchdown, and so forth. Table 1 shows the physical parameters, together with the values used in later simulations.

At stance phase, the dynamics is described as:

$$\begin{cases} M\ddot{r} + K_l(r - r_0) - Mr\dot{\theta}^2 = Mg(1 - \cos\theta) + f_l \\ J_l\ddot{\theta} + J_b\ddot{\phi} + \frac{d}{dt}(Mr^2\dot{\theta}) = rMg\sin\theta \\ J_b\ddot{\phi} + K_h(\theta - \phi) = \tau, \end{cases} \tag{18}$$

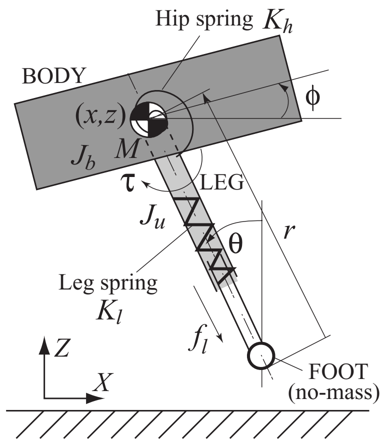

Fig. 2. Passive one-legged hopper

TABLE I
PHYSICAL PARAMETERS OF ONE-LEGGED MODEL

	Meaning	Unit	Value
g	gravity acceleration	m/s^2	9.8
M	total mass	kg	12
r_0	natural leg length	m	0.5
J_b	body inertia	kgm^2	0.5
J_l	equivalent leg inertia	kgm^2	0.11
K_l	leg spring stiffness	N/m	3000
K_h	hip spring stiffness	Nm/rad	10

where f_l is the control force of the leg, and τ is the control torque of the hip joint.

At flight phase, the dynamics is given by:

$$\begin{cases} \ddot{x} = 0 \\ \ddot{z} = -g \\ J_l\ddot{\theta} + J_b\ddot{\phi} = 0 \\ J_b\ddot{\phi} + K_h(\theta - \phi) = \tau, \end{cases} \tag{19}$$

At touchdown, the velocities of the generalized coordinates change instantaneously at touchdown phase by Assumption (C):

$$\begin{cases} \dot{x}_{td+} = \dot{x}_{td-} - \frac{J_l \cos\theta_{td}}{J_l + Mr_0^2}\mu_{td-} \\ \dot{z}_{td+} = \dot{z}_{td-} - \frac{J_l \sin\theta_{td}}{J_l + Mr_0^2}\mu_{td-} \\ \dot{\theta}_{td+} = \dot{\theta}_{td-} - \frac{Mr_0}{J_l + Mr_0^2}\mu_{td-} \\ \dot{\phi}_{td+} = \dot{\phi}_{td-} \\ \dot{r}_{td+} = \dot{z}_{td+}\cos\theta_{td} - \dot{x}_{td+}\sin\theta_{td}, \end{cases} \tag{20}$$

where

$$\mu_{td-} := \dot{x}_{td-}\cos\theta_{td} + \dot{z}_{td-}\sin\theta_{td} + r_0\dot{\theta}_{td-}. \tag{21}$$

At lift-off, there are no discontinuous changes of states except for:

$$\dot{r}_{lo} = 0. \tag{22}$$

B. Hybrid model representation

With the state variable $\underline{x} = [x,\ z,\ \theta,\ \phi,\ \dot{x},\ \dot{z},\ \dot{\theta},\ \dot{\phi}]^T \in R^8$, the above equations are summarized as:

- Continuous transitions (Eq. (18) and (19)):

$$\dot{\underline{x}}_i = f_i(\underline{x}) \quad (i = 1, 2) \tag{23}$$

- Discrete jumps (Eq. (20) and (22)):

$$\underline{x}_{i+} = h_i(\underline{x}_{i-}) \quad (i = 1, 2) \tag{24}$$

- Cross sections:

$$\Pi_1 \ := \ \left\{\underline{x} | S_1(\underline{x}) = z - r_0\sin\theta = 0\right\} \tag{25}$$
$$\Pi_2 \ := \ \left\{\underline{x} | S_2(\underline{x}) = \sqrt{x^2 + z^2} - r_0 = 0\right\} \tag{26}$$

Note that Eq. (18) can be expressed by (23) using the relationship $x = r\sin\theta$ and $z = r\cos\theta$. Also, the local coordinate $\underline{y}$ in Eq. (14) is defined as:

$$\underline{y} = \eta(\underline{x}) = [\ x \quad \theta \quad \phi \quad \dot{x} \quad \dot{z} \quad \dot{\theta} \quad \dot{\phi}\ ]^T \tag{27}$$

and its inverse map η^{-1} can be written as:

$$\underline{x} = \eta^{-1}(\underline{y}) = [\ x \quad r_0\cos\theta \quad \theta \quad \phi \quad \dot{x} \quad \dot{z} \quad \dot{\theta} \quad \dot{\phi}\ ]^T. \tag{28}$$

With this transformation, the dimension of Φ_{red} is (7×7) and search space becomes to 7. Furthermore, since we want to parameterize passive running gaits according to its forward speed $\dot{x}$, we can fix this parameter during the iteration of the Newton method. Then, the dimension of the search space finally reduces to 6.

C. Passive running gaits

Using the above formulation, the algorithm can be directly applied to the passive running model. The algorithm converges rapidly (approximately 10 iterations are sufficient until the error $< 10^{-8}$). The passive running gaits are found for every admissible initial guess and they are actually loss-less, i.e. no energy is dissipated.

Figure 3 shows the characteristic multipliers of passive running gaits with its forward speed varied from 0 to 3 m/s (Two trivial unity multipliers are not shown). For example, a fixed point $\underline{y}^* = [-0.1589, 0.3234, -0.0723, 2.0000, -2.0490, -2.4904, 0.5428]$ corresponds to a mid-speed passive running (2 m/s), whose stick animation is depicted in Fig. 4. The characteristic multiplier is calculated to $\mathrm{eig}(DP)$ = $[2.0098, -0.7682, -0.1230+0.1795i, -0.1230-0.1795i, 0.7382, 0.0000]$. On the other hand,

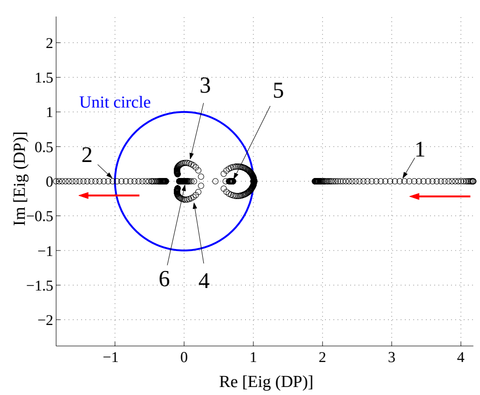

Fig. 3. Locus of the characteristic multipliers of passive running gaits, where the forward speed is varied from 0 to 3 m/s. The index numbers indicate the elements of the multipliers in order. Only the multipliers of vertical hopping (0 m/s) lie within the unit circle. Note that the second multiplier cross the unit circle, which indicates the bifurcation of periodic orbit.

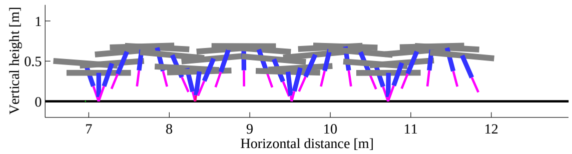

Fig. 4. Subsequent four steps of mid-speed passive running gait, where the running speed is 2 m/s

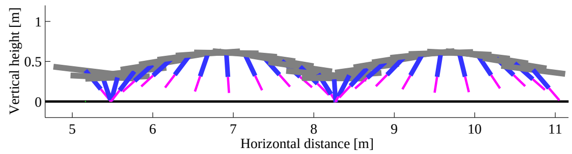

Fig. 5. Subsequent two steps of high-speed passive running gait, where the running speed is 5 m/s

a fixed point $\underline{y}^* = [-0.3129, 0.6763, -0.1198, 5.0000, -2.0760, -5.2004, 1.2818]$ corresponds to a high-speed passive running (5 m/s) (Fig. 5). The characteristic multiplier is calculated to $\mathrm{eig}(DP) = [\ 2.0755, -5.7708, -0.1933, 0.0220, 0.5940, 0.0000]$.

The stability of the gaits can be examined by Theorem 2.1. From Fig. 3, we can see bifurcations occur twice. At the speed 0 m/s, all the multipliers lie within and on the unit circle and the periodic orbit is *neutrally stable*. If the speed exceeds 0 m/s, a relatively large unstable multiplier appears in the right half plane. Then, the left multiplier crosses -1 when the speed exceeds 2.34 m/s. However, no remarkable differences between the gaits are observed (e.g. the period does not change). In summary, we can conclude the passive gaits are *orbitally unstable* (roughly speaking, although the robot starting from the fixed points can continue for several steps, it falls finally), except for trivial vertical hopping.

IV. Stabilization

Having obtained passive running orbits and found they are unstable, we must derive appropriate stabilizing controller. This is the most valuable stage of our approach to the energy-efficient legged locomotion (see Section I).

As the control inputs, suppose using the hip torqe τ at the flight phase only (for the reason see [16]). We consider the following piecewise constant input:

$$\tau = \begin{cases} \tau_1, & \text{if } 0 \le t < T_v/2 \\ \tau_2, & \text{if } T_v/2 \le t < T_v \end{cases}, \tag{29}$$

where τ_1 and τ_2 are the constant, and t indicates the time after the lift-off, and $T_v := 2\dot{x}_{lo}/g$ represents "expected" flight time.

A. Local stabilization by linear feedback controller

The first controller is a conventional local feedback controller. Linearizing the system around the unforced periodic orbits, we obtain the following closed-loop system:

$$\xi(k+1) = DP_\xi \xi(k) + DP_u \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}, \tag{30}$$

where $\xi := \underline{x} - \underline{x}^*$ is the error from the fixed points and DP_ξ is the same as the $DP_{\underline{x}}$ in the previous section. And $DP_u = \frac{\partial P}{\partial u} = (\frac{\partial P}{\partial \tau_1}, \frac{\partial P}{\partial \tau_2})$ is the newly appeared derivative due to the control inputs (29).

If at most two unstable multipliers can be made to be zero, all of the passive running gaits can be stabilized. This is actually possible because the closed-loop system is found to be *locally stabilizable* with the control input (29).

We do not show the simulation results here, because although the controller stabilizes all of the passive gaits, the region of attraction is found to be quite narrow. Actually, it is found by simulation that the local controller does not allow even 0.05 m/s error in initial velocity $\dot{x}_0$. If we use f_l as the inputs, the region of attraction could be enlarged [7], but the controller is local and strongly depends on $\underline{x}^*$.

B. Energy-Preserving Controller and Delayed-Feedback Controller

An alternative controller does not require priori knowledge about passive running gaits [12]. The central control strategy is *energy-preservation*, which aims to preserve the system energy as much as possible. We refer the reader to [16] for the details and give a brief outline here.

The essence of the controller is to design the touchdown angle θ_{td} and its velocity $\dot{\theta}_{td-}$ so as to meet the *non-dissipative condition*:

$$\mu_{td-} := \dot{x}_{td-} \cos\theta_{td} + \dot{z}_{td-} \sin\theta_{td} + r_0 \dot{\theta}_{td-} = 0. \tag{31}$$

This is a necessary condition for passive running because no energy is lost if this condition holds. For given translational lift-off velocities $(\dot{x}_{lo}, \dot{z}_{lo})$, there is a region of the pair $(\theta_{td}, \dot{\theta}_{td})$ that meets the condition. Delayed-Feedback Controller (DFC), widely used in chaos control [17], can give a good selection. If we consider the touchdown angle as a control inputs to the discrete dynamical system (return map), then we can use the adaptation law:

$$\overline{\theta}(k) = \begin{cases} -\frac{1}{2}\{\theta_{lo}(k) + \theta_{lo}(k-p)\}, & \text{if } k > p \\ -\theta_{lo}(k), & \text{else}, \end{cases} \tag{32}$$

where $\overline{\theta}(k)$ is the desired touchdown angle, and $p > 1$ is a desired period. That is, we try to control the period of the gaits by adapting touchdown angle. General form of discrete version of DFC and some discussion including its limitation, can be found in [18]. Having determined $\overline{\theta}$ and the associated desired velocity $\dot{\overline{\theta}}$, it is trivial to dead-beat second order linear system (19) by the control inputs (29).

Using this controller, "unknown" periodic gaits with "desired" period are orbitally stabilized. However, the control inputs does not disappear, hence the orbits do not converge to passive orbits. To solve this problem, in addition to the DFC, the following adaptive energy controller is applied:

$$f_l(k) = \gamma_f \cdot \mathrm{sign}\left[\Delta\tau(k-1)\right] \cdot \Delta\tau(k-1), \tag{33}$$

where $f_l(k)$ is the stepwise leg force, $\Delta\tau(k) := \tau_2(k) - \tau_1(k)$ means a "magnitude" of the control torque, and $\gamma_f > 0$ is an adaptation gain. Figure 6 shows one of the simulation results. From the figure, we can see the control inputs eventually converge to zero; perfect passive

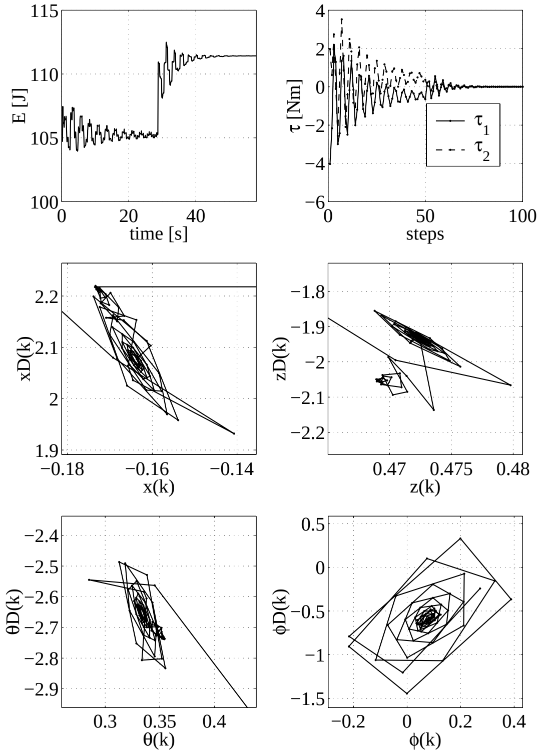

Fig. 6. Simulation results of the orbital stabilization to (unknown) 1-perioric gait and energy adaptation by leg force control: γ_f is set to 40 and the adaptation is activated from 60th step. E represents the total energy of the robot. Notation "D" represents the time derivatives of preceding variables.

running is achieved! From the adaptive update mechanism of Eq. (32), DFC can be considered as a kind of "real-time periodic orbits detector".

V. Conclusion

In this paper a new algorithm to detect hybrid periodic orbits of autonomous hybrid dynamical system was developed. The algorithm is very powerful tool and can be applied to any kinds of the robots (e.g. walking robot).

Conventional Newton algorithm was modified so that it can be applied to the analysis of return map of hybrid dynamical systems that include multiple phases (modes) and discrete jumps. With the algorithm, passive running gaits of one-legged robot were automatically detected for various parameter sets and initial conditions. In particular, the search space was reduced from $2n$ to $2n-1$ or less. This led to a great help for the analysis of the full-dimensional hybrid system behavior. The analysis of the characteristic multiplier of the return map revealed the stability and the bifurcation of the passive running gaits: all passive running gaits of one-legged robot were orbitally unstable except for a trivial vertical hopping gait. Two kinds of controllers that achieve orbital stabilization were presented; one was a local feedback controller based on the linearized return map, and the other was DFC-like like controller. Since the latter controller brought the system trajectories to the complete passive orbits, a similarity was found between the detection algorithm and the controller.

Our ongoing task includes exploring passive running gaits and stabilization of a biped and quadruped model [19], where two kinds of passive running gaits of planar quadruped were found. For biped robot with a torso, we have not found the passive running gaits so far. But, using the property of DFC as a "real-time periodic orbits detector", we have successfully obtained "unknown" stable periodic biped running gaits (not passive one) [20]. Since the controller is written in a state feedback form, if we express the closed-loop system as the autonomous hybrid dynamical system, then the algorithm is applicable and we will obtain "constrained periodic orbits", which will be presented near the future.

Acknowledgement

This research was partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Young Scientists, 15760144, 2003.

References

[1] M. Raibert, *Legged Robots That Balance*, MIT Press, 1985.

[2] D. Koditschek, M. Buehler, "Analysis of a simplified hopping robot", Int. J. Robotics Research, vol.10, no.6, pp.587–605, 1991.

[3] W. Schwind and D. Koditschek, "Control of forward velocity for a simplified planar hopping robot", Proc. of IEEE Int. Conf. on Robotics and Automation, pp.691–696, 1995.

[4] R. Altendorfer, D. E. Koditschek and P. Holmes, "Toward factored analysis of legged locomotion models", Proc. of IEEE Int. Conf. on Robotics and Automation, pp.37–44, 2003.

[5] C. Thompson and M. Raibert, "Passive dynamic running", Experimental Robotics I, eds. V. Hayward and O. Khatib., pp.74-83, Springer, 1989.

[6] M. Ahmadi and M. Buehler, "Stable control of a simulated one-legged running robot with hip and leg compliance", IEEE Trans. Robotics and Automation, vol.13, no.1, pp.96–104, 1997.

[7] C. François and C. Samson, "A new approach to the control of the planar one-legged hopper", Int. J. Robotics Research, vol.17, no.11, pp.1150–1166, 1998.

[8] S. Hyon, T. Emura and T. Ueta, "Nonlinear oscillations in legged running robots", IEICE NLP2004-9, pp.47–52, 2004 (in Japanese).

[9] T. Kousaka, T. Ueta and H. Kawakami, "Bifurcation of switched nonlinear dynamical systems". IEEE Trans. Circuits and Systems II, vol.CAS-II, no.7, pp.878–885, 1999.

[10] T. Parker and L. Chua, *Practical Numerical Algorithms for Chaotic Systems*, Springer, 1989.

[11] I. Hiskens and M. Pai, "Trajectory sensitivity analysis of hybrid systems", IEEE Trans. on Circuits and Systems - Part I: Fundamental Theory And Application, vol.47, no.2, pp.1057-7122, 2000.

[12] S. Hyon and T. Emura: "Quasi-periodic gaits of passive one-legged hopper, Proc. IEEE/RSJ Intelligent Robotics and Systems, pp.2625-2630, 2002.

[13] J. Guckenheimer and P. Holmes, *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Springer, 1983.

[14] J. Hale, *Ordinary Differential Equations*, Wiley, 1969.

[15] A. Isidori, *Nonlinear Control Systems: An Introduction*, 2nd ed. Springer, 1989.

[16] S. Hyon and T. Emura, "Energy-preserving control of passive one-legged running robot", Advanced Robotics, vol.18, no.4, 2004.

[17] K. Pyragas, "Continuous control of chaos by self-controlling feedback", Physi. Lett., A 170, pp.421–428, 1992.

[18] T. Ushio, "Limitation of delayed feedback control in nonlinear discrete-time systems", IEEE Trans. Circuits and Systems I, vol.43, pp.815–816, 1996.

[19] S. Hyon, X. Jiang, T. Emura and T. Ueta, "Passive running of planar 1/2/4-legged robots", Proc. IEEE/RSJ Intelligent Robotics and Systems, 2004 (to appear).

[20] S. Hyon and T. Emura, "Running control of a planar biped robot based on energy-preserving strategy", Proc. of IEEE Int. Conf. on Robotics and Automation, pp.3791–3796, 2004.