The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Follow-the-Contact-Point Gait Control of Centipede-Like Multi-Legged Robot to Navigate and Walk on Uneven Terrain

Shinkichi Inagaki, Tomoya Niwa and Tatsuya Suzuki

Abstract— **This paper proposes a novel locomotion control scheme of centipede-like multi-legged robot, which is called Follow-the-Contact-Point (FCP) gait control. A centipede-like multi-legged robot is composed of segmented trunks which have a pair of legs and are connected with fore and/or rear ones by joints. This control scheme realizes locomotion control of multi-legged robot on uneven terrain with perfectly decentralized manner. The main concept of the control scheme is to relay the contact points from the fore leg to the rear leg. By creating contact points of the first legs adequately on the environment, the robot can climb over obstacles and be navigated successfully. Finally, the result of physical simulation of a 20-legged robot shows the availability of the proposed method.**

I. INTRODUCTION

Multi-legged robots, which have more than six legs, have high locomotion ability and high fault tolerance thanks to high stability and redundant degrees of freedom. Centipede-like multi-legged robots as depicted in Fig.1 are expected to have good environment adaptability as well as snake-like robots in addition to the characteristic of multi-legged ones. However, the walking control is considered to be difficult because of the redundant degrees of freedom and realized only on even terrain in conventional researches [1], [2], [3].

The walking control of multi-legged robots is classified into three categories. One is based on implementation of mechanical impedance to legs (for example [4], [5]). This method can realize walking on uneven terrain by relatively simple control architecture and has characteristics of robustness to irregular disturbance. This implementation of mechanical impedance is considered to be a mimicry of mechanical feedback of arthropod, preflex [6].

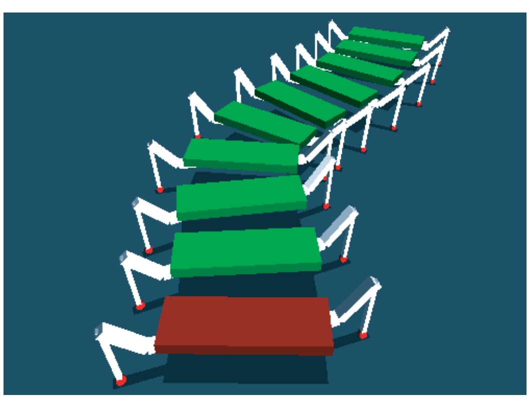

Fig. 1. Model of centipede-like multi-legged robot in physical simulator

This work was not supported by any organization

S. Inagaki, T. Niwa and T. Suzuki are with Dept. of Mechanical Science and Engineering, Nagoya University , Furo-cho, Chikusa-ku, Nagoya, Japan
inagaki@nuem.nagoya-u.ac.jp

Second is based on a model of nerve system dominating the walk control in animals (for example [7], [8]). In this method, adaptation to uneven terrain is achieved by sensory feedback corresponding to biological reflex. Especially, since the models of Central Pattern Generator (CPG) generally have a decentralized architecture, they are utilized to control decentralized and/or centipede-like multi-legged robots [2], [3], [9], [10].

The last is based on planning of motion. For example, Follow-The-Leader (FTL) gait control [11] are proposed for six-legged robots. FTL gait control is based on a strategy that the leg tips contact at the same points where the fore leg contacted. This method can achieve reliable performance of walking control on an uneven terrain, but is based on motion planning of the full-body motion. Therefore, it is difficult to apply centipede-type multi-legged robot because of the computational burden.

In this paper, we propose a new decentralized control for multi-legged robots, including centipede-like robots, to walk on an uneven terrain. This control law is called "Follow-the-Contact-Point (FCP) gait control" and categorized in the third of walking control architectures mentioned above. FCP gait control is inspired from a real centipede motion in which leg tips contact at the same points where the fore leg contacted [12]. This control scheme can yield gait generation and navigation simultaneously by planning contact points of the first legs. Although the concept idea is same as FTL gait control, FCP gait control realizes it by a perfectly decentralized control manner. In addition, the resultant gait generated by FCP gait control is definitely different from one of FTL gait control. This paper explains the algorithm of FCP gait control and the validity by showing some simulation results.

This paper is organized as follows: Section II introduces a multi-legged robot composed of N-segments. Sections III and IV cover control modes of each leg and the transition conditions of the control modes. Section V explains a method to allocate a new contact point of the first leg and adjust the stroke speed. Section VI reviews simulation results verifying the feasibility of our proposal and section VII presents conclusions.

II. DEFINITION OF SEGMENT AND CONTROL AREA

First of all, we define a segment consisting of a trunk and a pair of legs. The leg is composed of three links (Fig. 2). Each segment is connected serially with fore and/or rear segments

978-1-4244-6676-4/10/$25.00 ©2010 IEEE

through revolute passive joints, intersegment joints, of pitch and yaw angles. We hereinafter consider a multi-legged robot composed of $N(\geq 3)$ segments (Fig. 3). Left (l) and right (r) legs of segment $i \in \{1, \cdots, N\}$ have coordinate systems $C_i^s = (x_i^s, y_i^s, z_i^s)$, $s \in \{l, r\}$ respectively, and the origin is assigned to the beginning point of each leg (Link 1).

Each leg has a movable area M_i^s in which the leg tip can reach anywhere. We define a control area $A_i^s \subset M_i^s$ in which the leg tip is permitted to contact a terrain. In this paper, we define A_i^s as a cuboid whose dimensions are $L_x \times L_y \times L_z$. Note that the control area moves together with the trunk of the robot (see Fig. 8).

The control area of each leg overlaps ones of the fore and/or rear legs. Therefore, the control area A_i^s is divided into the following three or two areas:

Area 1: Area overlapping A_{i-1}^s.
Area 2: Area without overlap.
Area 3: Area overlapping A_{i+1}^s.

Note that the first and end legs do not have Area 1 and Area 3, respectively.

Hereinafter, if not otherwise specified, we omit upper suffix $s \in \{r, l\}$ of variables and parameters.

III. Control Modes of Leg

The leg motion is composed of two phases, swing phase, and stance phase. In the swing phase, the leg tip leaves the ground and approaches to the next contact point. In the stance phase, the leg tip keeps contact with the ground and makes a propulsive force. In FCP gait control, the both phases are controlled based on PD control as follows:

$$\tau_m = K_p(\hat{\theta}_m - \theta_m) - K_d \dot{\theta}_m \tag{1}$$

where τ_m is the torque of motor $m \in \{1, 2, 3\}$, θ_m the angle of motor m, $\hat{\theta}_m$ the desired angle of motor m which is calculated from inverse kinematics of desired position of the leg tip, $K_p(>0)$ and $K_d(>0)$ the proportional gain and the differential gain respectively. Leg control is composed of 4 modes, three control modes in the swing phase and one control mode in the stance phase (Fig. 4). The gains K_p and K_d are same in all control modes, but the desired motor angle $\hat{\theta}_m$ in (1) is switched in each mode. The reminder part of this section explains each control mode.

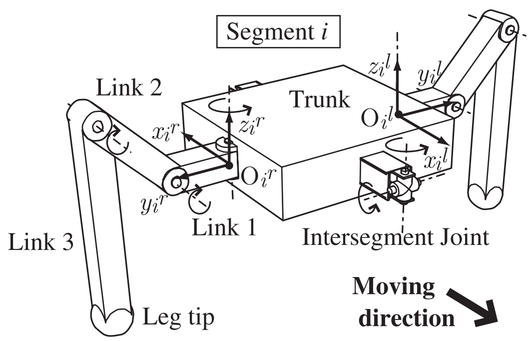

Fig. 2. Segment of centipede-like multi-legged robot

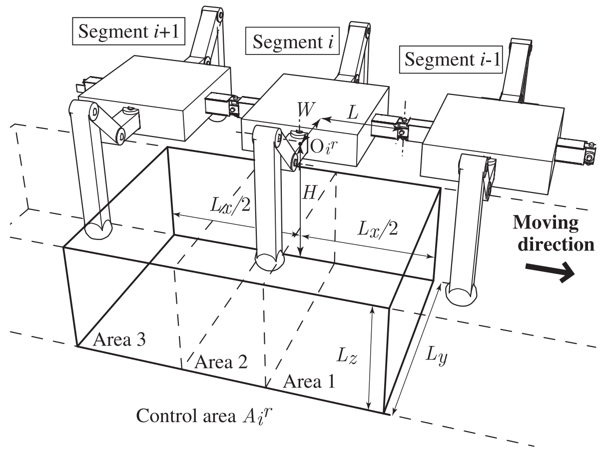

Fig. 3. Control area and overlap areas

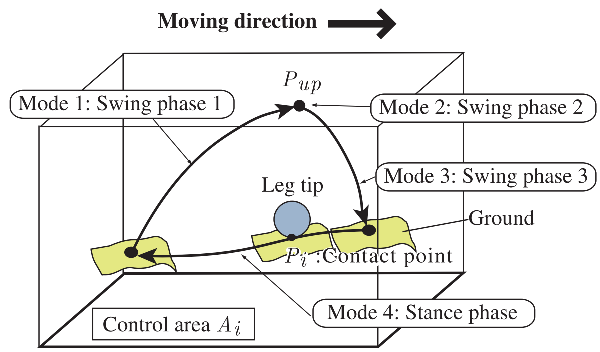

Fig. 4. Trajectory of leg tip in control area and corresponding control modes: Note that the contact point in stance phase is immovable in the absence of slipping. The control area does move with the trunk.

A. Control Mode 1: Swing Phase 1 (Leaving Contact Point)

In the swing phase 1, the leg tip leaves the contact point P_i and moves to a midair point P_{up} in the coordinate system C_i. The midair point P_{up} is allocated at the top of control area, $P_{up}^l = (X_{up}, Y_{up}, -Z_{up})$ in the left leg and $P_{up}^r = (-X_{up}, Y_{up}, -Z_{up})$ in the right leg. In this phase, the desired motor angle $\hat{\theta}_m$ in (1) is calculated from inverse kinematics of P_{up}.

B. Control Mode 2: Swing Phase 2 (Staying at Midair Point P_{up})

In the swing phase 2, the leg tip stays at the midair point P_{up}. This control is being activated until the leg tip of fore leg $i-1$ enters into Area 1 of leg i (see section IV for details). The desired motor angle $\hat{\theta}_m$ of (1) is calculated from inverse kinematics of P_{up} as same as the control mode 1.

C. Control Mode 3: Swing Phase 3 (Approaching to Next Contact Point)

In the swing phase 3, the leg tip leaves the midair point P_{up} and approaches to the next contact point $P_{new\,i}$ (Fig. 5). The position of the next contact point $P_{new\,i}$, except for the first legs ($i = 1$), is calculated based on the contact point of

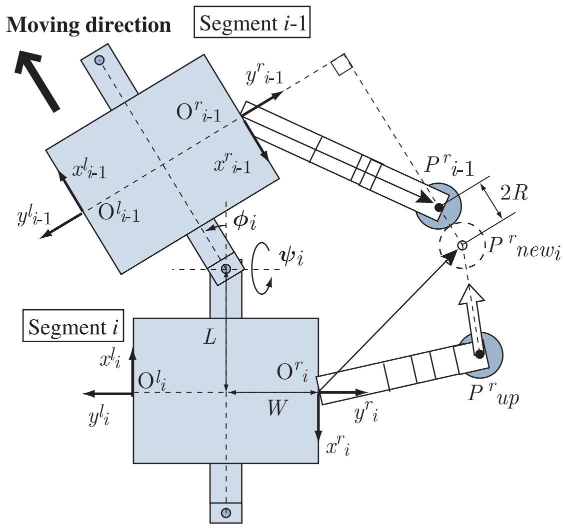

Fig. 5. Right leg of segment i in swing phase: The leg tip stays at the midair point P_{up}, and approaches the next contact point $P_{new\,i}$ by the control mode 3.

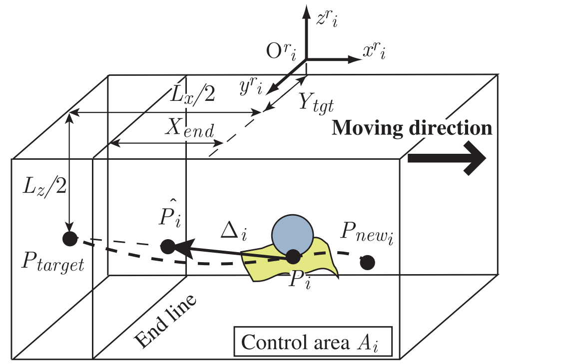

Fig. 6. Transition of contact point between two legs: The right leg of segment i is in stance phase. The leg tip P_i moves toward the target point P_{target} with contacting the ground.

the fore leg P_{i-1} in the left and right legs respectively as follows ($s \in \{l, r\}$, double-sign corresponds):

$$P^s_{new\,i} = R_1(i)R^s_2(i)\boldsymbol{w}^s(i) + \boldsymbol{b}^s(i), \tag{2}$$

$$R_1(i) = \begin{bmatrix} c\phi_i & -s\phi_i & 0 \\ s\phi_i & c\phi_i & 0 \\ 0 & 0 & 1 \end{bmatrix}, \; R_2(i) = \begin{bmatrix} c\psi_i & 0 & \mp s\psi_i \\ 0 & 1 & 0 \\ \pm s\psi_i & 0 & c\psi_i \end{bmatrix}, \tag{3}$$

$$\boldsymbol{w}(i) = \begin{bmatrix} x^s_{i-1} \mp 2R \pm L \\ y^s_{i-1} + W \\ z^s_{i-1} \end{bmatrix}, \; \boldsymbol{b}(i) = \begin{bmatrix} \pm L \\ -W \\ 0 \end{bmatrix} \tag{4}$$

where $P^s_{new\,i} = (x^s_{new\,i}, y^s_{new\,i}, z^s_{new\,i})^T$ and $P^s_{i-1} = (x^s_{i-1}, y^s_{i-1}, z^s_{i-1})^T$, $c\eta \equiv \cos\eta$ and $s\eta \equiv \sin\eta$, ϕ_i and ψ_i are respectively the yaw and pitch angles of the intersegment joint between trunks $i-1$ and i, $2W$, L and R are the width of trunk, the distance between the joint center and the trunk center, and the radius of leg tip, respectively. The next contact point of the first segment $P_{new\,1}$ is given by searching the environment autonomously or by an operator directly.

The timing to leave the leg tip from P_{up} is determined based on the position of the fore leg tip ($i \geq 2$) or the position of the produced next contact point $P_{new\,1}$ ($i = 1$) (see section IV for details). In this phase, $\hat{\theta}_m$ of (1) is calculated from inverse kinematics of the next contact point $P_{new\,i}$.

D. Control Mode 4: Stance Phase

Once the leg i contacts the next contact point $P_{new\,i}$, that is $P_i = P_{new\,i}$, the desired contact point $\hat{P}_i$ is calculated at each control sampling instant as follows (Fig. 6):

$$\hat{P}_i = P_i + \frac{(P_{target} - P_i)}{|P_{target} - P_i|}\Delta_i \tag{5}$$

where Δ_i is the propulsion distance of the desired contact point, P_i the current contact point, P_{target} the target point, $|P_A - P_B|$ means the length between points P_A and P_B. The coordinate of the target point P^l_{target} is $(L_x/2,\ Y_{tgt},\ L_z/2)$ in the left leg and P^r_{target} is $(-L_x/2,\ Y_{tgt},\ L_z/2)$ in the right leg, respectively. The propulsion distance Δ_i adjusts the velocity of the leg tip. Different propulsion distances between the left and right sides is capable of increasing the turning performance of the robot. The value of propulsion distance Δ_i inherits the fore leg value Δ_{i-1} when the leg i contacts the next contact point $P_{new\,i}$.

The timing of the leg tip leaving from the contact point is determined according to the rear and ipsilateral legs except for the end legs. As for the end legs, the leg tip leaves the contact point when it crosses End line which is assigned at $x = \pm X_{end}$ in the left leg and the right leg, respectively. In this phase, $\hat{\theta}_m$ of (1) is calculated from inverse kinematics of $\hat{P}_i$.

IV. Transition of Control Mode

Follow-the-Contact-Point (FCP) gait control is represented by Mealy automata (Fig. 7). Mealy automaton can represent a reactive system which produces output events according to input events. The state transition is described as follows:

$$\text{inEvent[guard]/outEvent;}\ \cdots\ ;\ \text{outEvent} \tag{6}$$

where "inEvent" is an input event and "guard" is a guard condition. When the inEvent occurs and the guard condition is true, the mode transition fires and output events "outEvent" are produced simultaneously. When the inEvent is omitted, the transition can fire anytime. The transition with inEvent has higher priority than one without inEvent. The transition conditions are checked each control sampling instant.

The state of automata of FCP gait control corresponds to each control mode described in section III. Since the first ($i = 1$) and end ($i = N$) legs have no fore and rear legs respectively, the transition conditions are different from ones of middle legs. The reminder part of this section explains about each automaton.

A. Automaton of Middle Leg

The state 1 corresponds to the swing phase 1 of leg i, in which the leg tip is approaching the midair point P_{up} by the

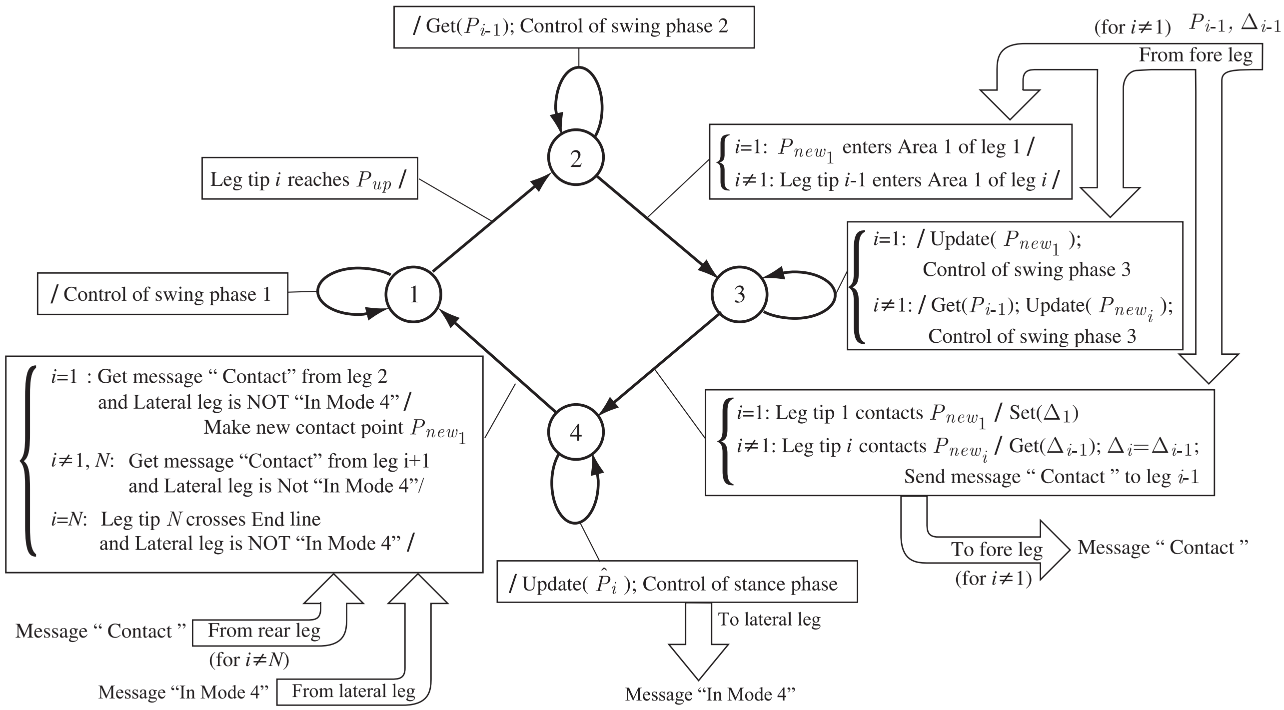

Fig. 7. Automaton of middle leg ($i \neq 1, N$), first leg ($i = 1$) and end leg ($i = N$) in FCP gait control

control mode 1. When the leg tip reaches the midair point P_{up}, the state changes to the state 2.

In the state 2, the leg is controlled so as to keep the leg tip at the midair point P_{up} by the control mode 2. Also, in this state, the leg gets coordinate of the fore leg tip P_{i-1} to calculate (2), which is represented as "Get(P_{i-1})" in Fig. 7. When the leg tip of fore leg $i-1$ enters Area 1 of leg i, the state changes to the state 3.

In the state 3, the leg gets coordinate of the fore leg tip P_{i-1} and calculates the new contact point P_{newi} based on (2), which is represented as "Update(P_{newi})" in Fig. 7. The leg is controlled by the control mode 3 so that the leg tip approaches the new contact point P_{newi}. When the leg tip contacts the new contact point P_{newi}, the leg i inherits the propulsion distance of the desired contact point Δ_{i-1} from the fore leg and sends a message, "Contact" which means "The leg tip contacted the ground," to the fore leg $i-1$.

In the state 4, the desired position of the leg tip $\hat{P}_i$ is calculated based on (5) at every control sampling instant and the leg tip is controlled to follow the desired point. A message "In Mode 4" is always sent to the ipsilateral leg over this state in order to avoid the both legs being the swing phase simultaneously. When the leg receives a message "Contact" from the rear leg $i+1$ and does not receive a message "In Mode 4" from the ipsilateral leg, the state changes to 1 again.

Figure 8 represents the moving control areas and the mode changes of two legs. The control areas move with the trunks, but the contact point does not move on the ground. FCP gait is achieved by transferring the contact points from the first leg to the end leg.

Note that FCP gait control is perfectly decentralized since each segment only needs neighbor information:

- Coordinate of the fore leg tip P_{i-1},
- Propulsion distance of desired contact point Δ_{i-1},
- Messages from the rear leg $i+1$ and the ipsilateral leg.

B. Automaton of First and End Leg

Since the first leg ($i = 1$) has no fore leg, it has to make a new contact point P_{new1} by itself. A new contact point P_{new1} is produced when the control mode changes from mode 4 to 1. The method to produce a new contact point is described in section V.

On the other hand, since the end leg ($i = N$) has no rear leg, it has to decide the timing when the leg tip leaves the contact point P_N. The leg tip leaves the contact point when the leg tip crosses End line (Fig. 6).

V. New Contact Point of First Leg

A. Allocation of New Contact Point

A new contact point of the first legs P_{new1} is made ahead of the robot. The new contact point enters the control area of the first leg at some time when the robot moves forward. By a new contact point allocated according to the desired moving direction and on an obstacle, the robot can change the moving direction and climb over the obstacle, respectively.

In this paper, as for change of the moving direction, the positions of new contact points are derived empirically as shown in Fig. 9. In addition, when an obstacle exists ahead of the robot, the new contact point is made on the obstacle as shown in Fig.10. An operator gives the robot three commands

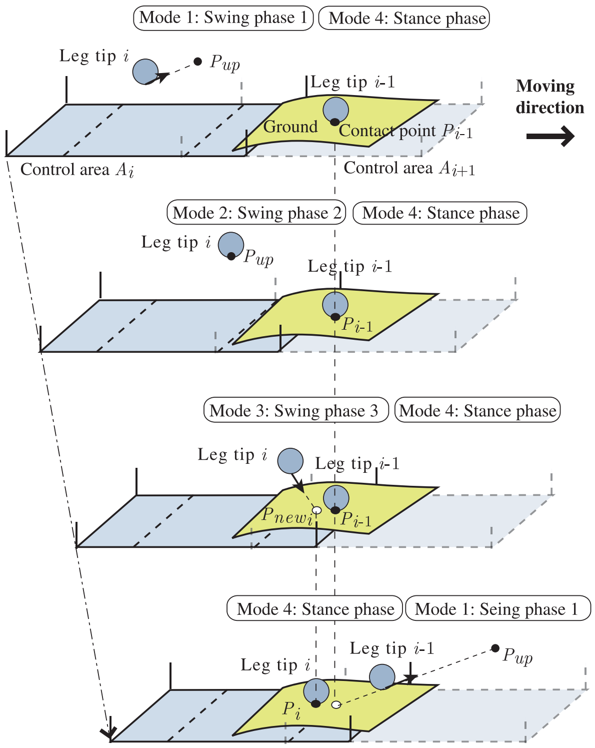

Fig. 8. Moving control area and mode transition

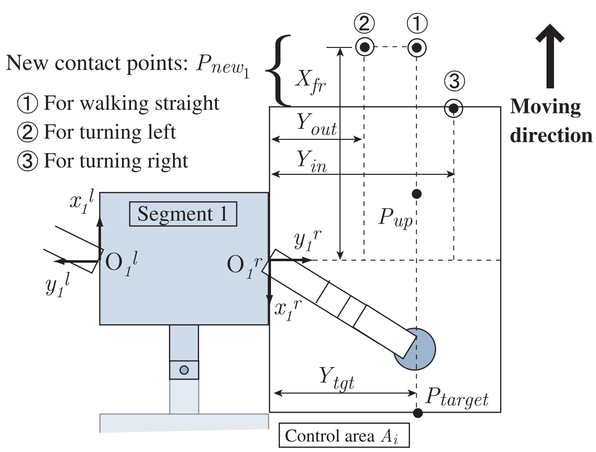

Fig. 9. New contact point of first leg to control moving direction

of moving direction, going forward, turning right and turning left. When moving forward, a new contact point is allocated on the same line of the target point P_{target} with some distance ahead from the front edge of the control area. When turning left (right), a new contact point is allocated on the left (right) side of the control area. As the result, the segment gets a rotational torque for turning.

The stroke length of the leg motion depends on the position of a new contact point P_{new1}. Since the position affect the locomotion performance, it has to be selected adequately. Also, autonomous production of new contact points of the first legs by searching the environment is our future work.

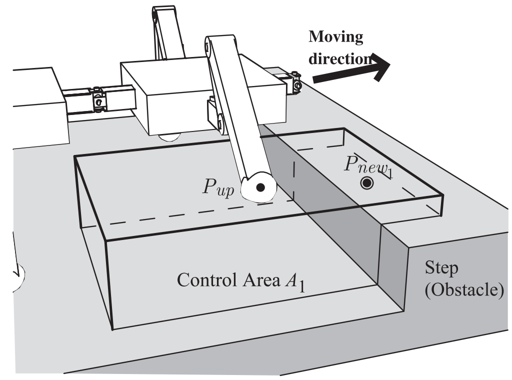

Fig. 10. New contact point of first leg to get over obstacle

B. Adjustment of Stroke Speed

The segment can get rotational torque by different stroke speed between the left and right legs. When the left leg is faster (slower) than the right leg, the segment rotate to right (left). The stroke speed in stance phase is adjustable by changing the propulsion distance of the desired contact point Δ_i. Since Δ_i is inherited from the fore leg $i-1$, the moving direction of the robot can be controlled simply by changing the propulsion distance of the first legs Δ_1. In actual control of moving direction, combination of the new contact point allocation and the stroke speed adjustment is adopted.

VI. SIMULATION

We used three-dimensional physical simulator PhysX [www.nvidia.com] to verify our proposed gait control, FPC gait control. PhysX can simulate dynamics of large-scale systems in real time. We combined rigid bodies, joints, and motors to construct a 10 segments, 20-legged, robot in the simulation environment. We made the trunks and the legs from aluminum, the leg tips from rubber and the ground and the obstacle from asphalt, and set the density, the reflection coefficient and the static and dynamic friction coefficients to plausible values. Each leg is controlled by an individual automaton described in section IV. The sampling time of the gait controller was 10 [ms] and one of dynamics calculation was 1 [ms]. The gravity acceleration was 9.81[$\mathrm{m/s^2}$].

A. Robot Configuration and Parameters

Figure 1 shows the multi-legged robot which is constructed in the simulation environment. The component dimensions of the segment are shown in Table I. Parameters used in FCP gait control are derived empirically (Table II). k_p and d_p are the elastic and viscous coefficients of the pitch angle of the intersegment joint, respectively. In addition, k_y and d_y are ones of the yaw angle. In the initial state, the control mode of each leg was set to 4 and the coordinate of each leg tip was $(0,\ Y_{tgt},\ L_z)$.

B. Simulation Result

Figure 11 shows the operation performance of the robot. In order to control the moving direction, we used the

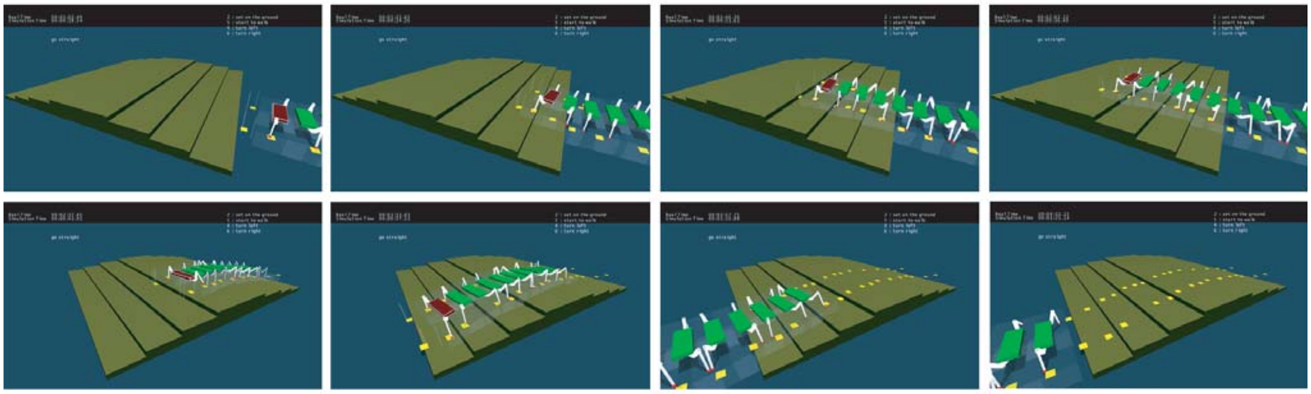

Fig. 12. Simulation result: Climbing over stairs

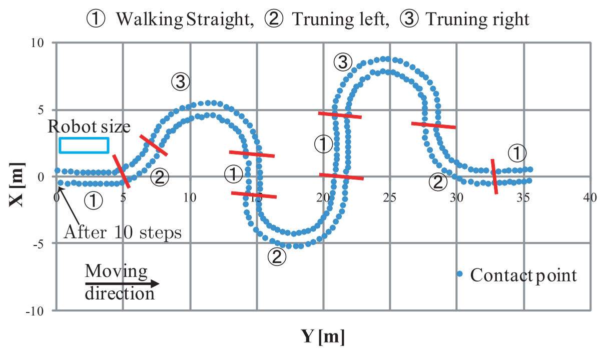

Fig. 11. Operation performance: Dots are contacting points of first legs

combination of the new contact point allocation and the stroke speed adjustment. When the robot walked straight, the propulsion distance of the desired contact point Δ_1 was set 0.02 [m] in both left and right sides. When turning left, Δ_1 was set 0.04 [m] in the right side and 0.01 [m] in the left side. When turning right, the opposite values were set. An operator gave three commands, "walk straight," "turn left," and "turn right." The plotted points on the ground are contact points of the left and right first legs.

Figure 12 shows the robot climbing over steps. The height of one step was 0.2 [m]. In this experiment, the first segment sometimes has bended to the right or the left side because of the passivity of the intersegment joint. Introduction of active intersegment joints is also our future work.

TABLE I
DIMENSIONS

Part	Size (w×d×h) [m] Mass [kg]
Trunk	0.40 × 0.20 × 0.05 3.0
Link 1	0.2 × 0.04 × 0.04 0.9
Link 2	0.2 × 0.04 × 0.04 0.9
Link 3	0.3 × 0.02 × 0.02 0.35
Leg tip	Radius [m]= 0.02 0.017

TABLE II
PARAMETERS [MKS]

N	10	R	0.035
K_p	600	Y_{tgt}	0.375
K_d	10	X_{end}	0.2
L_x	0.6	H	0.1
L_y	0.5	X_{fr}	0.5
L_z	0.4	Y_{in}	0.35
X_{up}	0.05	Y_{out}	0.3
Y_{up}	0.4	k_p	150
Z_{up}	0.1	k_y	10
W	0.15	d_p	10
L	0.175	d_y	10

VII. CONCLUSION

We proposed Follow-the-Contact-Point (FCP) gait control to control multi-legged robots walking on a level terrain. FCP gait control is decentralized and event-driven control which can be represented by automata. We showed the validity by some simulation results using a physical simulator. Parameter optimization by using automaton verification and developing a real centipede-like multi-legged robot are our another future works.

REFERENCES

[1] A.Torige, S.Yagi, H.Makino, T.Yagami, N.Ishizawa "Centipede type walking robot (CWR-2)," *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.402–407, 1997.

[2] S.Aoi, H.Sasaki, and K.Tsuchiya "A multilegged modular robot that meanders: investigation of turning maneuvers using its inherent dynamic characteristics," *SIAM Journal on Applied Dynamical Systems*, Vol.6, No.2, pp.348–377, 2007.

[3] L.Matthey, L.Righetti and A.J.Ijspeert "Experimental study of limit cycle and chaotic controllers for the locomotion of centipede robots," *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.1861–1865, 2008.

[4] S.Skaff et al.: "Inertial Navigation and Visual Line Following for a Dynamical Hexapod Robot," *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Vol.2, pp.1808–1813, 2003.

[5] D.E.Koditschek, R.D.Full, M.Buehler: "Mechanical aspects of legged locomotion control," *Arthropod Structure and Development* , Vol33. pp.251–272, 2004.

[6] M.H.Dickinson, C.T.Farley, R.J.Full, M.A.R.Koehl, R.Kram and S.Lehman "How Animals Move: An Integrative View," *Science*, Vol.288, pp.100–106, 2000.

[7] K.S.Espenschied, R.D.Quinn, R.D Beer and H.J,Chiel "Biologically based distributed control and local reflexes improve rough terrain locomotion in a hexapod robot," *Robotics and Autonomous Systems*, pp. 59–64, 1996.

[8] H.Cruse, Th.Kindermann, M.Schumm, J.Dean, J.Schmitz "Walknet–a biologically inspired network ti control six-legged walking–," *Neural Networks*, 11, 1435-1447, 1998.

[9] T.Odashima, H.Yuasa, M.Ito "The autonomous decentralized myriapod locomotion robot which is consist of homogeneous subsystems," *JRSJ*, Vol.16, No.7, pp.81–88, 1998.

[10] S.Inagaki, T.Suzuki, H.Yuasa, T.Arai "Wave CPG model for autonomous decentralized multi-legged robot –gait generation and walking speed control–," *Robotics and Autonomous Systems*, Vol.54, pp.118–126, 2006.

[11] F.Ozguner, S.J.Tsai, R.B.McGhee "An approach to the use of terrain-preview information in rough-terrain locomotion by a hexapod machine," *The International Journal of Robotics Research*, Vol.3, No.2, pp.134–146, 1984.

[12] B.D.Anderson, R.J.Full, T.Chen, "Mechanics of locomotion in centipedes," *American Zoologist*, Vol.30, No.4, p.135A, 1990.