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

Head-navigated Locomotion of a Snake-like Robot for Its Autonomous Obstacle Avoidance

Xiaodong Wu and Shugen Ma

***Abstract*— Inspired from the natural motion of snakes, a head-navigated locomotion is proposed for snake-like robots in this paper. Different from the traditional serpentine motion, the head of the snake-like robot would always maintain its direction along the forward direction of the whole robot. By utilizing this particular characteristic, the obstacles in front of the undulatory motion could be detected conveniently by a distance sensor installed on the head module. Based on the analysis of the configuration of the snake-like robot in the head-navigated locomotion, CPG-based control is introduced for the control system. Two kinds of control strategies for the realization of the automatic obstacle avoidance are proposed. Both simulation and experiment have been carried out to verify the proposed motion pattern.**

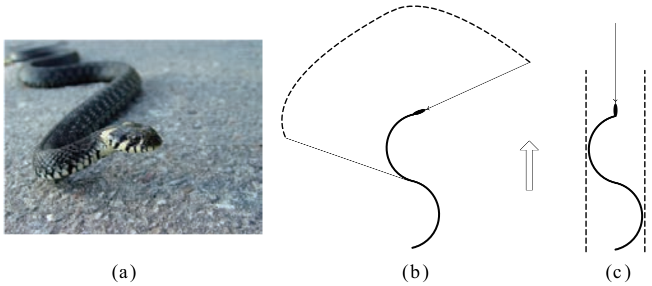
(a) (b) (c)

Fig. 1. Bio-inspired head-navigated locomotion. (a) Natural snake. (b) Detection area in the traditional *S*-shape locomotion. (c) Detection area in the head-navigated locomotion

I. INTRODUCTION

The snake-like locomotive mechanism has long been studied for its capability in moving on complicated and unknown terrain. In recent years, extensive research has been carried out to realize an environment-adaptive locomotion for a snake-like robot. Some of them are based on ingenious mechanisms designed to adapt to the surroundings passively, including wheel-less snake-like robots for an obstacle-aided locomotion in [1][2], and a snake-like robot utilizing passive joints in [3]. Others, which are also dominant, rely on an operator to control the robot actively to adapt to the unstructured environment. A semi-autonomous snake robot with 3 axial force sensor has been developed in Hirose's laboratory [4]. An energy-based control method for the snake-like robot to adapt to different friction and slope has also been proposed in [5]. Crespi and Ijspeert have taken online optimization of swimming and crawling in an amphibious snake robot by CPG method [6].

To allow autonomously adapted motion of the snake-like robot to the surroundings, a successful navigation of robot with obstacle avoidance is integrant. The sensory information of obstacle is necessarily considered into the control system for the robot to move in unpredictable environments. Due to the particular structure and *S*-shape motion pattern, the snake-like robot can not turn right or left as simple as wheeled robots. The design of the control system to avoid barriers should consider both the motion pattern and environment together. Researchers have proposed many kinds of solutions for the navigation of snake robot locomotion. Most of them used sensory information to construct a closed-loop controller. Paap and Christaller [7] have realized a semi-automatic motion of the snake robot by scanning the environment with a video camera and two ultrasonic sensors. Sfakiotakis and Tsakiris [8] have used numbers of distance sensors to collect the information of surroundings and navigate an undulatory robot in a complex corridor. Similarly, Caglav [9] also used IR range detectors in a caterpillar-like robot. By the utilization of tactile sensors on each segment of snake robot, adaptive motion has been performed by Umetani [10]. Position control of a snake-like robot by trajectory tracking has been studied in [11].

In this paper, a head-navigated motion pattern of snake-like robot is proposed for obstacle avoidance. This head-navigated motion is inspired from the natural snake which always lifts its head to detect the front area during the movement, which is shown in Fig. 1 (a). Despite use of an *S*-shape locomotion, it can be found that the head of the snake keeps the same orientation during forward locomotion. In most research, the snake-like robot moves in an *S*-shape pattern where the head of robot is swinging from side to side. The estimation of obstacle information will be difficult because the forward direction of movement can not be decided conveniently. For example, if a distance sensor is mounted on the head of the snake-like robot, the detecting area can be plotted like that in Fig. 1 (b). This irregular fan-shaped detection area will make it difficult for the robot to judge which object is a true obstacle that affects the robot motion and which one is not. However, compared with traditional locomotion, the detection area of the head-navigated motion as shown in Fig. 1 (c), is just along the forward direction of the robot movement. By simply adding one sensor on the head of the robot, the effective obstacle information can be obtained easily. In [8], they also used

---

This work was partly supported by the National Natural Science Foundation of China under Grant 60875083

X. Wu is with Department of Robotics, Ritsumeikan University, 525-8577 Shiga, Japan `gr041087@ed.ritsumei.ac.jp`

S. Ma is with Department of Robotics, Ritsumeikan University, 525-8577 Shiga, Japan; and Shenyang Institute of Automation, CAS, China `shugen@se.ritsumei.ac.jp`

distance sensors to collect environmental information for the navigation of an undulatory robot. However, a large number of sensors have been mounted from head to tail on the robot body. Thus the calculation and signal processing were complicated and time-consuming. For this reason, if a bio-inspired head-navigated motion pattern can be introduced into the snake-like robot for autonomous obstacle avoidance, it will show its advantages.

This paper is organized as follows. Some features and assumptions of the collision-free behavior of the snake-like robot will be described in section 2. A simple approach to realize proposed head-navigated locomotion will be explained in section 3. In section 4, based on the previous research in [12], the CPG-based control method for real-time obstacle avoidance is presented. In section 5, the design of autonomous obstacle avoidance for the snake-like robot would be analyzed, and experiments on a real snake-like robot are also conducted to show the effectiveness of the locomotion control. Finally, the work of this paper is concluded in section 6.

II. COLLISION FREE BEHAVIOR

We first define a set of desirable features for our locomotion controller. Some assumptions on the controller are also proposed to limit our design consideration.

A. Features

Generally, robot navigation problems can be classified as global or local ones [13] [15]. In global navigation, the information of the environment surrounding the robot is known and a path to pass through the obstacles is decided beforehand. Graphical map is one example of the global navigation techniques. In [14], terrain maps are integrated into a reactive navigation strategy of field mobile robots operating on rough terrain. On the other hand, in local navigation, the environment surrounding the robot is unknown, or only partially known, and a collision avoidance controller has to be incorporated into the robot to avoid the obstacles detected by the sensors. The artificial potential field approach is one of the well-known methods which has been used for the local obstacle avoidance [13].

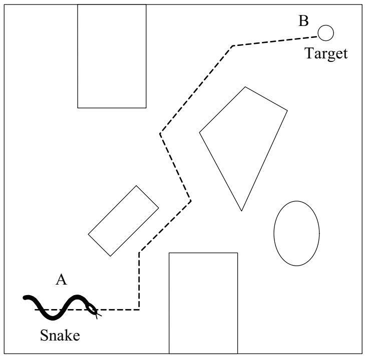

Fig. 2. Global and local navigation of the snake-like robot

In the design of the control system for the snake-like robot, we divide it into two levels: high level control for global navigation and low level for local navigation. For example, the snake-like robot is planned to move from point A to B as shown in Fig. 2. A high level command from human or computer will give a selected route to guide the robot's movement toward the target. However, the detailed information of the obstacles can not be assured. When the snake robot comes across an obstacle, it will depend on the low level control to realize local navigation automatically. This paper focuses on the local navigation problem which means that the geometry and the location of the obstacles are unknown. Based on the real-time detection by the distance sensor, the snake-like robot can decide to move in three directions, forward, left or right. The turning angle of the robot's movement is determined by the sensory signal of the obstacle. The advantages of the proposed scheme are that less local information is required than other techniques and the navigation law is simpler and more flexible.

B. Assumptions

For simplicity, we only consider the snake-like robot moving on flat ground. This means that the obstacle avoidance is just on the two-dimensional horizontal plane. Furthermore, the snake-like robot has one IR sensor to measure the distance between itself and the obstacle. Some assumptions are made for the obstacles and the distance sensor as follows.

(1) The shapes of all obstacles are supposed to have flat surface.

(2) The size of obstacle should be larger than the width of the shape motion.

(3) The distances between the obstacles are larger than the measurable distance of the IR sensor.

(4) The measurable distance of the IR sensor is at least equal to the forward length of the snake-like robot in one period.

Assumptions (1) and (2) simplify the environmental model for the analysis of motion control. Assumption (3) is to avoid the "dead lock" of the locomotion where the robot may be blocked up. Assumption (4), as will be explained later, is considered to have enough time for turn motion.

III. HEAD-NAVIGATED LOCOMOTION

Inspired from the natural snake, the head of the snake-like robot would be controlled to lift up and keep the same orientation as the forward direction. A control scheme is needed to make the robot maintain the direction of its head during undulating locomotion. Yamada in [16] has studied the stabilization of the head of an undulating snake-like robot. A sin-based continuous model and discrete model have been analyzed elaborately for the motion control. Herein, due to the different CPG-based control model, we will give another approach to realize the head control of the snake-like robot.

The concept of the head-navigated locomotion is shown in Fig. 3. The blue squares represent the head of the snake-like robot during the motion. Despite the *S*-shape motion pattern, the head module can always point to the direction of the forward movement. The dash line means the detection

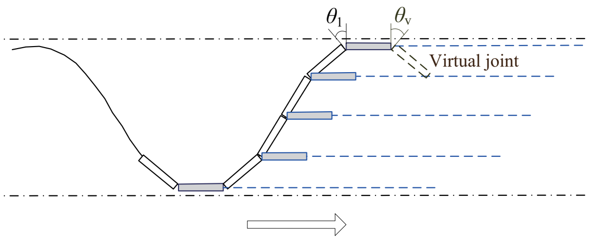

Fig. 3. Head-navigated locomotion of the snake-like robot

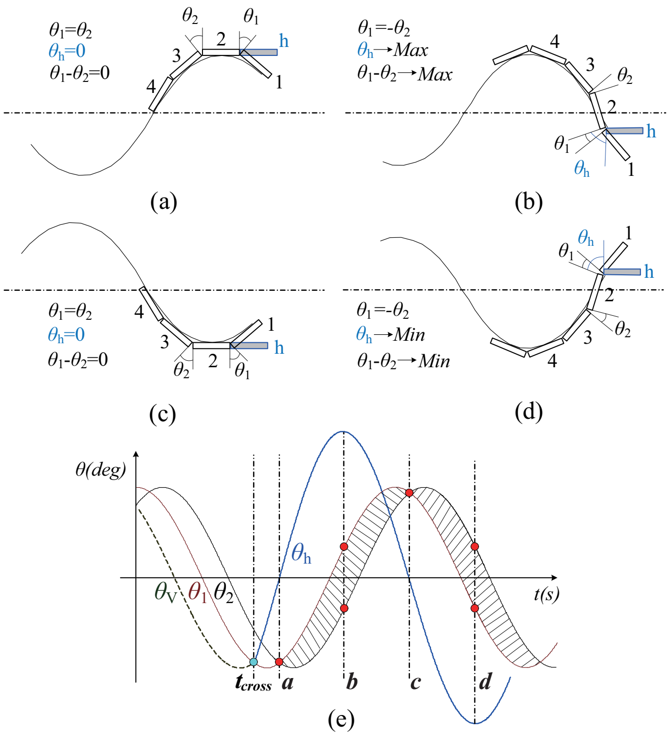

Fig. 4. Configurations of the joint angles in the head-navigated locomotion of the snake-like robot

area of a distance sensor which is installed on the head. It can be found that the sensory information in this kind of locomotion will just explore the front area which the robot will pass through.

In the traditional serpentine locomotion, the angle of head joint also swings from side to side just as the other joints on the robot body. To obtain a head-navigated locomotion, the angle of the first joint which connects the head and second link should be controlled by a totally different model. There are three basic features of the angle of the first joint:

(1) The direction of the head module should always be kept the same as the forward direction of robot.

(2) The angle of the first joint is also changed periodically and has an identical frequency as the body joints.

(3) The angle of the first joint has the same kind of periodic signal as the body joints but different phase and amplitude.

The turning point from the original head control to the head-navigated control is the problem that we are concerned with. As shown in the Fig. 3, we add a virtual joint before the head, and set the angle of it having the same phase difference with the first joint as that between the first joint and the second one. By using this virtual joint as a reference, when the head module is located at the situation where $\theta_1 = \theta_v$, the direction of head will just be facing the forward direction of the snake-like robot. Furthermore, by analyzing the relation between the original head joint angle and the desired head joint angle, four typical situations of the joints during the head-navigated locomotion have been shown in Fig. 4.

In the Fig. 4, θ_1 is the original angle of the first joint under normal *S*-shape locomotion; θ_2 is the angle of the second joint; θ_h is the modified angle of the first joint under head-navigated locomotion. In *situation (a)*, where the original value of first joint θ_1 is the same as second joint θ_2, θ_h should be zero to let the head link point to the forward direction. Meanwhile, $\theta_1 - \theta_2$ is also zero at this time. In *situation (b)*, where the original value of first joint θ_1 just equals the inverse value of second joint θ_2, θ_h comes to its maximal value. For the value of $\theta_1 - \theta_2$ also get to the Maximum. Due to the feature (2), the configuration of the joints in situation (c) and (d) are similar with that of situation (a) and (b) which are just different in phase. These four typical situations compose a whole period of locomotion. The relations of each angle signals are clearly shown in Fig. 4 (e). From these results, it is obvious to find that the angle signal of θ_h has the same phase and frequency as those of $\theta_1 - \theta_2$. Since the same kind of periodic signals as stated in feature (3), the angle signal of θ_h can be expressed as a function of $\theta_1 - \theta_2$. If we define the original function of the angle of the first joint as $f(t)$, and phase difference of two adjacent joints as φ, the equation of desired head joint angle θ_h could be described as follows:

$$\begin{aligned} \theta_1 &= f(t) \\ \theta_v &= f(t+\varphi) \\ \theta_2 &= f(t-\varphi) \\ \theta_h &= A(\theta_1 - \theta_2) = A[f(t) - f(t-\varphi)] \end{aligned} \tag{1}$$

From the above equations, the major problem in control of the head joint has changed to solve the value of amplitude coefficient A. From Fig. 4 (e), the crossing time t_{cross} of θ_1 and θ_v can be calculated easily. Meanwhile, due to θ_1 equals θ_h at this time, the value of amplitude coefficient A can be obtained by:

$$A = f(t_{cross})/[f(t_{cross}) - f(t_{cross} - \varphi)] \tag{2}$$

IV. CPG-BASED CONTROLLER DESIGN

The rhythmic creeping motion of a natural snake is generated by the CPG mechanism. The CPG neural oscillators in the spinal cord stimulate the muscle extension and contraction to generate rhythmic swing of the body. By constructing an oscillation network mimicking the neural system of the animal, a series of successive rhythmic signals with the certain phase difference can be generated to realize the snake-like locomotion. Fig. 5 shows a CPG network implemented to control a snake-like robot, where one CPG

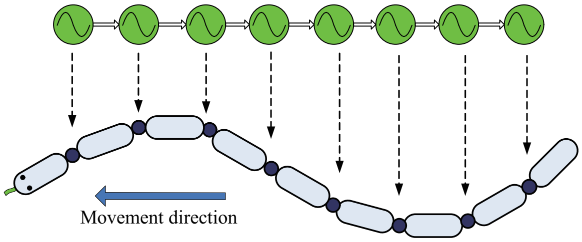

Fig. 5. CPG network implemented to control a snake-like robot

module corresponds to one joint motor. Each CPG works as a basic neural oscillator to generate rhythmic signals for the angle of the robot joint.

A. Mathematical Model of CPG Network

Based on our previous work in [5], a closed-loop CPG network with feedback connection is adopted for the control of a snake-like robot in this paper. A CPG network includes n CPG modules which have m neurons can be described in a group of basic equations. For the j-th neuron of the i-th CPG module, its mathematical model can be described by

$$\begin{aligned}
&\tau_{1,i}\dot{u}_{j,i} + u_{j,i} = u_{0,i} - \beta v_{j,i} - w y_{s,i} + \sum_{k=1}^{n} w_{ik} y_{j,k} \\
&\tau_{2,i}\dot{v}_{j,i} + v_{j,i} = y_{j,i} \\
&y_{j,i} = g(u_{j,i}) = max(0, u_{j,i}) \\
&y_{out,i} = y_{1,i} - y_{2,i} \\
&i, k = 1, 2, ...n, \;\; i \neq k; j = 1, 2, ...m; \\
&s = \begin{cases} m, & \text{if } j = 1 \\ j-1, & \text{others} \end{cases}
\end{aligned} \tag{3}$$

where n is the number of CPG modules in the network; m is the number of neurons in one CPG module; s is the serial number of neuron connected to the j-th neuron; $u_{j,i}$ is the membrane potentials of j-th neuron in the i-th CPG module; $v_{j,i}$ is the variable that represents the degree of adaptation; $u_{0,i}$ is the tonic driving input; $\tau_{1,i}$ and $\tau_{2,i}$ are the parameters that specify the time constants for membrane potential and adaptation degree, respectively; β is the adaptation coefficient; w is the weight between neurons; w_{ik} is the connection weight of the i-th module from the k-th module; $y_{j,i}$ is the output of j-th neuron in i-th CPG module; $y_{out,i}$ is the output of the i-th CPG module.

B. CPG-controlled Head-navigated Locomotion

Based on the analysis of the head-navigated locomotion in section III, the configuration of the first joint in the snake-like robot needs to be redesigned separately. A virtual CPG is needed to be added for the judgment of the start point of head-navigated motion. From the equation (1) and equation (2), the control signal of the head joint $y_{out,head}$ is calculated from the output of the first CPG $y_{out,1}$ and the second CPG $y_{out,2}$ given in equation (4). A is an amplitude coefficient which can be calculated from the crossing point of $y_{out,1}$, $y_{out,virtual}$ and $y_{out,head}$ at time t_{cross}. One simple demonstration of the control signals from the CPG network is shown in Fig. 6.

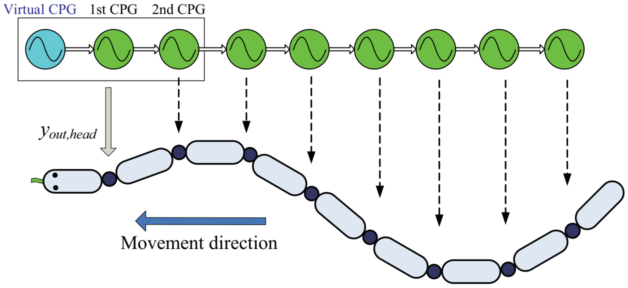

Fig. 6. CPG network implemented to control a head-navigated locomotion

$$\begin{cases}
y_{out,head}(t) = A[y_{out,1}(t) - y_{out,2}(t)] \\
y_{out,1}(t_{cross}) = y_{out,virtual}(t_{cross}) \\
y_{out,1}(t_{cross}) = y_{out,head}(t_{cross}) \\
A = y_{out,1}(t_{cross})/[y_{out,1}(t_{cross}) - y_{out,2}(t_{cross})]
\end{cases} \tag{4}$$

When the snake-like robot meets with the barrier in the head-navigated locomotion, a turn motion to avoid the obstacle should be performed. In the previous research [12], a parameter transmitting principle of the CPG network has been used for the real-time control of the turn motion. The adjustment of the CPG configuration will transmit one by one from the head segment to the rear with a fixed shift interval. To realize turn motion of the robot, the driving input u_0 of CPG have been used to adjust the amplitude of body wave from the head to the tail.

In this paper, a similar method has been introduced. When the obstacle is detected by the head module, a modified signal of the driving input will be generated and transmitted into the control system. Here the head module equipped with sensors, is worked as a control center to conduct the navigation autonomously.

C. Simulation

In order to verify the proposed control method for the head-navigated locomotion, a simulation has been performed in Open Dynamics Engine (ODE) environment. The rhythmic output of the CPGs are implemented on the joints of the robot as the angle input signals. The simulation results are shown in the Fig. 7. The locomotion in the 1st and 2nd scenes is still a traditional serpentine movement where the head module is swinging from the side to side. The 3rd scene is the turning point where head module just has the same direction as that of forward movement. From this point, the motion will change into head-navigated locomotion smoothly, and maintain its orientation as shown in 4th and 5th scenes. The control signals of joint angles are shown in Fig. 8. The dash-and-dot line between the red line and the blue line represents the turning point from the original head motion to the head-navigated motion, which corresponds to the 3rd scene of Fig. 7.

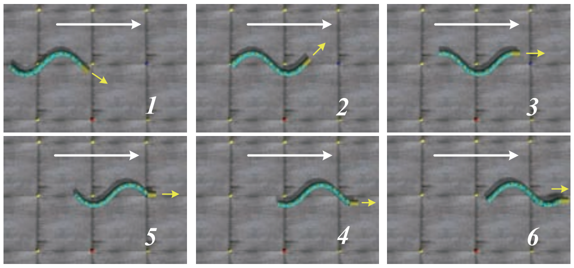

Fig. 7. Simulation results of the head-navigated locomotion

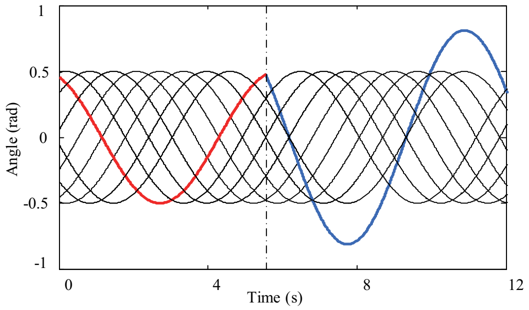

Fig. 8. Control signals of the joints in the simulation. Red line and blue line represent the angle signal of the first joint in two different motion patterns. Black lines are the angle signals of other joints.

V. AUTONOMOUS OBSTACLE AVOIDANCE

Sensory information needs to be coupled into a control system to realize autonomous obstacle avoidance. Under the head-navigated locomotion, the distance between the head of the snake-like robot and the obstacle can be obtained by IR sensor. How to utilize this information to conduct the avoidance behavior will be discussed in this section.

A. Control Strategy

From the above description, the control strategy will be based on the assumptions of the environmental model proposed in section II. Some parameters of the snake-like robot when meeting with a obstacle have been shown in Fig. 9 (a). If the angle between the surface of the obstacle and the forward direction of robot can be detected, a corresponding turning angle would be decided for automatic avoidance. The distance between the head and the obstacle at peak points of the S-shape L_1 and L_2 can be detected by IR sensor. S_L and S_A are the basic configuration of robot body which can be measured beforehand. Thus the value of desired turning angle can be calculated by following function:

$$\theta_{turn} = \tan^{-1}\left(\frac{S_A}{L_1 - L_2 - S_L}\right) \tag{5}$$

When the turn angle has been decided, the robot can avoid the obstacle by altering the parameters of CPG network. The detailed process can refer to our previous research [12]. This method needs an accurate distance signal from the sensor. The value of S_L and S_A should also be remeasured when the configuration of body shape is changed. For this reason, another control strategy is proposed. The basic illustration of this strategy is shown in Fig. 9 (b). The snake-like robot will keep turning until the IR sensor can not detect any obstacle or the obstacle is out of its measuring range. However, this method is a semi-autonomous control where a command to decide left turning or right turning is still needed.

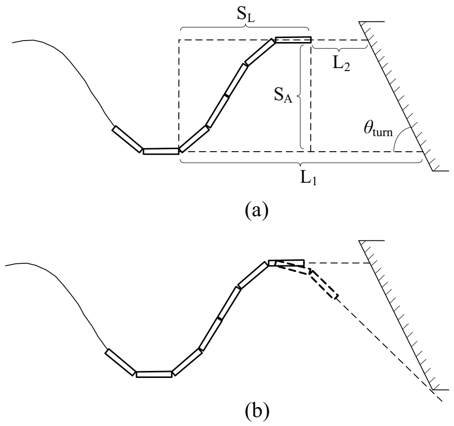

Fig. 9. Two control strategies for automatic obstacle avoidance

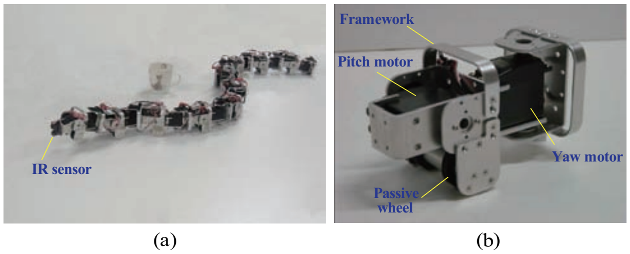

Fig. 10. (a) Overview of the snake-like robot (b) Mechanical structure of one module

B. Experiment

To testify the validity of CPG-controlled obstacle avoidance, some experiments were carried out on our snake-like robot. The robot model is composed of 20 joints which are connected serially. The overview and mechanical structure of the robot are shown in Fig. 10. At the head of the robot, an IR sensor is mounted to detect the obstacle in front of the forward movement. The specifications of the robot are shown in table I.

TABLE I

MECHANICAL SPECS OF THE SNAKE-LIKE ROBOT

Specifications	Details
Number of joints :	20
Size of link[mm]:	130×62×77
Weight of link[kg]:	0.28
Motion range of yaw angle[deg]:	[-90, +90]
Actuators:	RC servo motor (FUTABA S3305)
Sensor:	IR sensor (SHARP GP2Y0A710K0F)

By implementing rhythmic outputs derived from the CPG-based control system onto the driving joints, the snake-like robot successfully exhibits head-navigated locomotion

Fig. 11. Experiment scenes of the head-navigated locomotion

as shown in Fig. 11. In the experiments, the first scene shows the motion of the robot without head-navigated control. From the second scene, the head begins to keep its direction which is the same as the whole direction of the movement.

When the information of the distance sensor is coupled into the control system, an automatic obstacle avoidance is performed by the snake-like robot as shown in Fig. 12. It can be found that a continuous change of the locomotion curve for the turn motion is achieved without any human command.

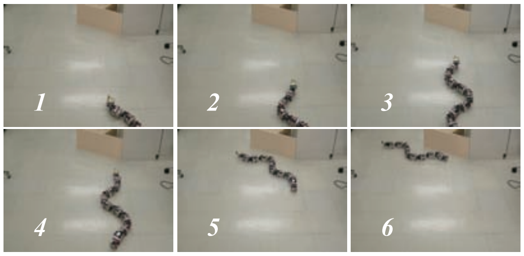

Fig. 12. Experiment scenes of the autonomous obstacle avoidance

Herein, a further experiment with more obstacles has not been carried out. Due to the low accuracy of IR sensor (±10cm error), the snake-like robot can not always obtain exact turning angle to avoid the obstacle. Thus a high-precision sensor or modified algorithm will be considered in future. Furthermore, the slip of the passive wheels in the normal direction also influences the stable motion of the snake-like robot. The forward direction can not be always kept as the desired direction. Nevertheless, the proposed theory has shown its validity in the snake-like locomotion.

VI. CONCLUSIONS AND FUTURE WORKS

In this paper, a head-navigated locomotion which is inspired from the motion of the natural snake has been proposed. In this kind of motion pattern, the head of snake-like robot would maintain its direction just as that of the whole forward motion. From this particular characteristic, the obstacles in front of the undulatory body could be detected conveniently just by a distance sensor installed on the head module. From the analysis of the configuration of the snake-like robot in the head-navigated locomotion, the control model for the head joint has been established. CPG-controlled turn motion has been introduced for the robot to realize the avoidance behavior autonomously. Compared with other navigation methods, there are following advantages in this head-navigated locomotion: (1) Only one sensor is needed to detect the obstacle and the signal processing is simplified markedly; (2) An automatic obstacle avoidance can be realized in real-time. Finally, both simulation and experiment have been carried out to verify the proposed motion method.

As stated in this paper, the design has been limited to a local navigation of the robot. To deal with a complex situation, more information of the environment needs to be collected. Thus a global or a high level controller should be considered to couple with the local navigation in future, which could improve the performance and efficiency of the navigation.

REFERENCES

[1] A.A. Transeth, R.I. Leine, C. Glocker and K.Y. Pettersen, 3-D Snake Robot Motion: Nonsmooth Modeling, Simulations, and Experiments. *IEEE Transactions on Robotics*, vol.24(2), 2008, pp. 361-376.

[2] Z.Y. Bayraktaroglu, Snake-like locomotion: Experimentations with a biologically inspired wheel-less snake robot. *Mechanism and Machine Theory*, vol.44(3), 2009, pp. 591-602.

[3] K. Ito and Y. Fukumori, Autonomous control of a snake-like robot utilizing passive mechanism, *In Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Orlando, USA, 2006, pp. 381-386.

[4] S.R. Taal, H. Yamada, and S. Hirose, "3 Axial Force Sensor for a Semi-Autonomous Snake Robot", *In Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan, 2009, pp. 4057-4062.

[5] X. Wu and S. Ma, Adaptive creeping locomotion of a CPG-controlled snake-like robot to environment change, *Autonomous Robots*, vol. 28(3), 2010, pp. 283-294.

[6] A. Crespi, and A.J. Ijspeert, Online Optimization of Swimming and Crawling in an Amphibious Snake Robot, *IEEE Transactions on Robotics*, vol.24(1), 2008, pp. 75-87.

[7] K.L. Paap, T. Christaller, and F. Kirchner, "A Robot Snake to Inspect Broken Buildings", *In Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2000, pp. 2079-2082.

[8] M. Sfakiotakis and D.P. Tsakiris. Biomimetic Centering for Undulatory Robots, *The International Journal of Robotics Research*, vol.26(11-12), 2007, pp. 1267-1282

[9] E. Caglav, A.M. Erkmen, and I. Erkmen, "A snake-like robot for variable friction unstructured terrains, pushing aside debris in clearing passages", *In Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, USA, 2007, pp. 3685-3690.

[10] Y. Umetani, and S. Hirose, "Biomechanical study of active cord mechanism with tactile sensors", *In Proceedings of International Symposium on Industrial Robots*, Nottingham, UK, 1976, pp. c1-1-c1-10.

[11] F. Matsuno, H. Sato, "Trajectory Tracking Control of Snake Robots Based on Dynamic Model", *In Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 3029-3035.

[12] X. Wu and S. Ma, CPG-controlled Asymmetric Locomotion of a Snake-like Robot for Obstacle Avoidance, *In Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics*, Guilin, China, 2009, pp. 69-74.

[13] R.B. Tilove, "Local obstacle avoidance for mobile robots based on the method of artificial potentials", *In Proceedings of the 1990 IEEE International Conference on Robotics and Automation*, Cincinnati, USA, 1990, pp. 566-571.

[14] A. Howard, B. Werger, and H. Seraji, "Integrating Terrain Maps into a Reactive Navigation Strategy", *In Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, Taipei, Taiwan, 2003, pp. 2012-2017.

[15] A. Fujimori, P.N. Nikiforuk, and M.M. Gupta, Adaptive Navigation of Mobile Robots with Obstacle Avoidance, *IEEE Transactions on Robotics and Automation*, vol. 13(4), 1997, pp. 596-602.

[16] H. Yamada, M. Mori, and S. Hirose, Stabilization of the head of an undulating snake-like robot, *In Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, USA, 2007, pp. 3566-3571.