2003 International Symposium on Advanced Intelligent Systems
September 25-28, 2003, Jeju, Korea

# A Study of the Obstacle Avoidance for a Quadruped Walking Robot Using Genetic and Fuzzy Algorithm

Bo-Hee Lee*, Jung-Shik Kong** and Jin-Geol Kim***
* Department of Electrical Engineering, Semyung University
** Department of Industrial Automation, Inha University
***School of Electrical Engineering, Inha University
Email : bhlee@semyung.ac.kr

**Abstract— This paper presents the leg trajectory generation for the quadruped robot with genetic-fuzzy algorithm. To have the mobility even at uneven terrain, a robot is able to recognize obstacles, and generates moving path of body that can avoid obstacles. This robot should have its own avoidance algorithm against obstacles, forwarding to target without collision. During walking period, a robot recognizes obstacle from external environment with a PSD and some interface, and this obstacle information is converted into proper the body rotation angle by fuzzy inference engine. After this process, we can infer the walking direction and walking distance of body, and finally can generate the optimal leg trajectory using genetic algorithm. All these methods are verified with PC simulation program, and implemented to SERO-V robot.**

## I. INTRODUCTION

The human-interactive robots gives people some pleasure and aids for the disabled person at indoors, outdoors, uneven terrain, and even at stairway. To do this job, it is required to design the leg type of human-interactive robots with sensor system, which can recognize the uneven terrain and obstacles. Among this robot, a quadruped robot is easy to realize because of easiness to sustain stability. Recently, these concepts have been implemented to such as Sony's AIBO[1,2] and so on.

Generally, important issues for designing a quadruped robot are the recognition ability for environment and free walking ability. To move various real life environment even at uneven terrain, the obstacle recognition method from external environment is required [3]. Also to walk freely, a robot must have the ability that can changes body movement according to via points of leg, and research for path generation of the body is suggested [4].

But relationship from leg trajectory to body path generation can't be expressed clearly, also optimal leg trajectory of quadruped robot can't be defined clearly. Therefore, to guarantee free walking ability of quadruped, optimal body path generation and smooth leg motion generation are needed simultaneously.

In this paper, we propose a fuzzy algorithm to utilize generating the avoidance path against the obstacle. From sensor system, robot recognizes front obstacles. If obstacle is not, robot may move straightly, otherwise obstacles are exist, it should determine the avoidance path for the obstacle, moving into the target point. Fuzzy algorithm helps the robot determine the optimal path to avoid obstacles, and optimal gait trajectory from genetic algorithm is generated in order to make smooth leg's motion.

## II. QUADRUPED ROBOT CONSTRUCTION

### *A. System Configuration*

We designed a quadruped robot, called SERO-V, which have 14 joints and a main controller. Each joint is actuated by RC-motor and main controller is an Atmega103 8-bit microprocessor from ATMEL. It deals with a PSD sensor, motor driving, and gives serial interface with PC. Every several time intervals, the main controller communicates with a PC simulator, and sends some recognition signal, receive each angle data of quadruped robot respectively. Fig 1 shows the configuration of SERO-V.

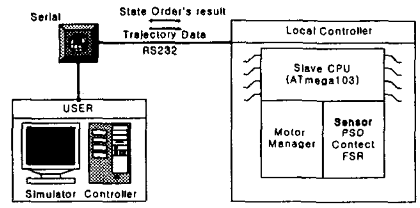
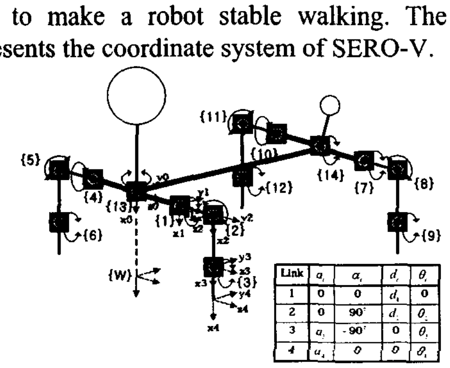



Fig. 1. Total configuration of SERO V

### *B. Body and legs analysis*

Our robot has the total 14 motors including 3 joints each leg and one at head and one at tail. For the part of leg, two of the motors are used as the shoulder joint to realize walking action, and the other performs a knee joint to make a robot stable walking. The Fig. 2 represents the coordinate system of SERO-V.




| Link | $a_i$ | $\alpha_i$ | $d_i$ | $\theta_i$ |
|---|---|---|---|---|
| 1 | 0 | 0 | $d_1$ | 0 |
| 2 | 0 | 90° | $d_2$ | $\theta_2$ |
| 3 | $a_3$ | -90° | 0 | $\theta_3$ |
| 4 | $a_4$ | 0 | 0 | $\theta_4$ |

Fig. 2. Coordinate system of SERO IV

Because robot legs have the symmetry structure, we tried to do kinematics analysis only a front-left leg, and the others can be expanded easily. The inverse kinematics of a front-left leg is as follows.

$$\theta_2 = \sin^{-1}\left(\frac{-\beta \pm \sqrt{\beta^2 - \alpha\gamma}}{\alpha}\right) \tag{1}$$

$$\alpha = p_x^2 + p_y^2 \qquad \beta = a_4 p_y s_4 \qquad \gamma = a_4^2 s_4^2 - p_x^2 \tag{2}$$

$$\theta_3 = \sin^{-1}\left(\frac{2ka_3}{p_x^2 + p_y^2 - a_4^2 + a_3^2 + (p_z - d_1 - d_2)^2}\right) \tag{3}$$

$$\theta_4 = \cos^{-1}\left(\frac{p_x^2 + p_y^2 - a_4^2 + a_3^2 + (p_z - d_1 - d_2)^2}{2a_3 a_4} - \frac{a_3}{a_4}\right) \tag{4}$$

Where, $P_x$, $P_y$, $P_z$, $a_3$, $a_4$, $d_1$ and $d_2$ are the position vectors of the end points and link parameters, $\theta_2$ is the shoulder's pitch joint, $\theta_3$ is the shoulder's yaw joint, and $\theta_4$ is the knee joint, respectively. Inverse kinematics of the others leg can be derived by only changing the base coordinate.

## III. 3. Obstacle Avoidance Algorithm

To generate path that can avoid obstacles, a robot has to recognize obstacles from external environment. In order to obtain obstacle information, a robot is installed a PSD sensor and an orientation motor.

At first, a robot checks relationship between its body and front obstacle. The relationship data are specified with two parameters, one is the shortest distance $d$ to the front direction obstacle, and the other is the shortest width $w$ that the robot will avoid front obstacles without collision (Fig. 3). After finding the distance and angle for the obstacle, the robot should find the body projection angle to move. We use fuzzy algorithm to find the proper direction and angle. Generally speaking, fuzzy algorithm is valid to find the optimal solution quickly with robustness [5].

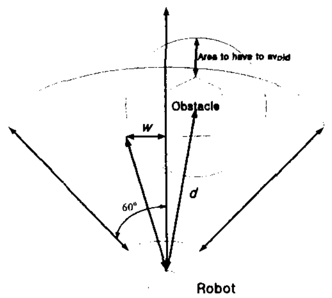



Fig. 3. Definition of environments

Table 1 shows rule-base of the fuzzified input angle to avoid obstacles. In Table 1, from D-4 to D+4 are the fuzzified short width( $w$ ) to be able to avoid front obstacle. from E-3 to E+3 mean the fuzzified distance( $d$ ) between robot and obstacles respectively.

Table 1. $w, d$ Rule-base

| | E-3 | E-2 | E-1 | E0 | E+1 | E+2 | E+3 |
|---|---|---|---|---|---|---|---|
| D-4 | -9 | -7 | -6 | -5 | -4 | -3 | -3 |
| D-3 | -8 | -6 | -5 | -4 | -4 | -3 | -2 |
| D-2 | -7 | -5 | -4 | -3 | -3 | -2 | -2 |
| D-1 | -6 | -3 | -3 | -2 | -2 | -2 | -1 |
| D0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D+1 | 5 | 3 | 3 | 2 | 2 | 2 | 1 |
| D+2 | 6 | 5 | 4 | 3 | 3 | 2 | 2 |
| D+3 | 7 | 6 | 5 | 4 | 4 | 3 | 2 |
| D+4 | 9 | 7 | 6 | 5 | 4 | 3 | 3 |

We use 56 rule-bases, and the numbers cost the available body propulsion angles range from –90 to +90 degree. From this rule-base, the robot can select optimal body direction to avoid obstacle.

## IV. Gait Generation Algorithm

In this paper, to make smooth walking, we apply genetic algorithm. Generally speaking, a genetic algorithm is used in searching optimal result. It has good performance searching optimal solution [6-8]. It can find the optimal solution without solving any differential equation for the dynamic system. Genetic algorithm makes new generation with the series string data represented the regeneration, crossover and mutation, which parameters are shown in Table 2.

Table 2 Parameters of GA

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| Generation No. | 250 | Population No. | 100 |
| Crossover Rate | 0.7 | Mutation Rate | 0.2 |
| String No. | 240 | | |
| Fitness function | $\dfrac{1}{\sum(\dot{\theta}_{i+1} - \dot{\theta}_i)^2 + \sum(\ddot{\theta}_{i+1} - \ddot{\theta}_i)^2}$ | | |

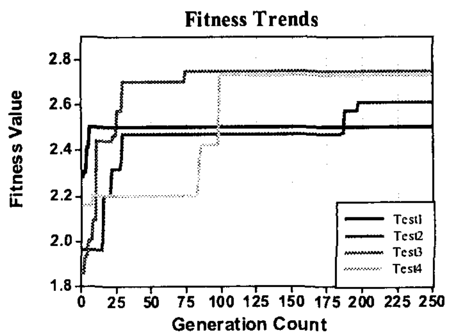



Fig.4. Fitness trends of front-left leg

All of the parameters such as a crossover rate and mutation rate are determined by many experiments. Where, $\theta$, $\dot{\theta}$, $\ddot{\theta}$ are angle, angular velocity and angular acceleration of each leg joint, and fitness function is derived from the minimization of movement of each leg joint, and The string number of genetic algorithm is represented by via points count and each axis. The Fig. 4 shows the fitness trends according to the generation times, which number of generation times is determined by experiments. The Fig. 5 is an overall flowchart of the trajectory generation based on fuzzy-genetic algorithm, represented from getting environment data to finding the proper body angle and position.

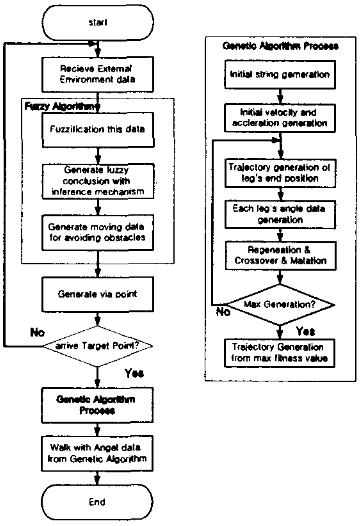



Fig. 5. Flowchart of fuzzy-genetic algorithm

From obstacle avoidance algorithm, robot can obtain body movement data. Body movement data has distance and angle to walk. This data are not converted legs' via point directly. To match body moving data and via point of leg, we limit and simplify legs movement. Fig. 6 is leg movement flow each times. Because at least three legs of quadruped robot have to remain on the surface to maintain stable walking, the determination of legs movement according to period of time is important. At each time, via points are determined by the method of the Fig. 6. In Fig. 6, the numbers mean end points of leg each times, and Table 3 is via points with respect to walking distance and angle, where $l$ is distant to want to move, $\theta$ is desired direction, and sub script number of this table means the instance of body propulsion with respect to time.

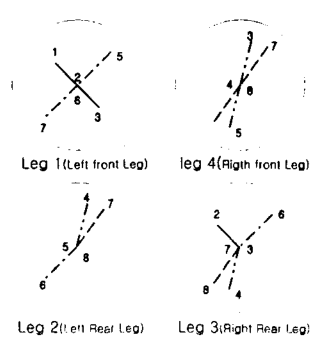



Fig. 6. Plain view of legs according to walking .

Tab. 3 legs' via points w.r.t. position and orientation of body

| | Leg1 | | | Leg4 | | |
|---|---|---|---|---|---|---|
| | x | y | Z | X | y | z |
| 1 | $l_1\cos\theta_1$ | 0 | $l_1\sin\theta_1$ | $-l_0\cos\theta_0$ | 0 | $-l_0\sin\theta_0$ |
| 2 | 0 | 0 | 0 | 0 | -0.05 | 0 |
| 3 | $-l_1\cos\theta_1$ | 0 | $-l_1\sin\theta_1$ | $l_2\cos\theta_2$ | 0 | $l_2\sin\theta_2$ |
| 4 | 0 | -0.05 | 0 | 0 | 0 | 0 |
| 5 | $l_3\cos\theta_3$ | 0 | $l_3\sin\theta_3$ | $-l_1\cos\theta_1$ | 0 | $-l_1\sin\theta_1$ |
| 6 | 0 | 0 | 0 | 0 | -0.05 | 0 |
| 7 | $-l_3\cos\theta_3$ | 0 | $-l_3\sin\theta_3$ | $l_4\cos\theta_4$ | 0 | $l_4\sin\theta_4$ |
| 8 | 0 | -0.05 | 0 | 0 | 0 | 0 |
| | Leg2 | | | Leg3 | | |
| | x | y | Z | X | y | z |
| 1 | 0 | 0 | 0 | 0 | -0.05 | 0 |
| 2 | $-l_2\cos\theta_2$ | 0 | $-l_2\sin\theta_2$ | $l_1\cos\theta_1$ | 0 | $l_1\sin\theta_1$ |
| 3 | 0 | -0.05 | 0 | 0 | 0 | 0 |
| 4 | $l_3\cos\theta_3$ | 0 | $l_3\sin\theta_3$ | $-l_2\cos\theta_2$ | 0 | $-l_2\sin\theta_2$ |
| 5 | 0 | 0 | 0 | 0 | -0.05 | 0 |
| 6 | $-l_4\cos\theta_4$ | 0 | $-l_4\sin\theta_4$ | $l_3\cos\theta_3$ | 0 | $l_3\sin\theta_3$ |
| 7 | 0 | -0.05 | 0 | 0 | 0 | 0 |
| 8 | $l_5\cos\theta_5$ | 0 | $l_5\sin\theta_5$ | $-l_4\cos\theta_4$ | 0 | $-l_4\sin\theta_4$ |

In fact, the values of table 4 are via points of legs during walking, and these are calculated by heuristic method added on triangular function.

## V. Simulation Result

All of the algorithms are developed with Visual C++ in PC environment. We design a PC based commander for the algorithm as shown in Fig. 7.

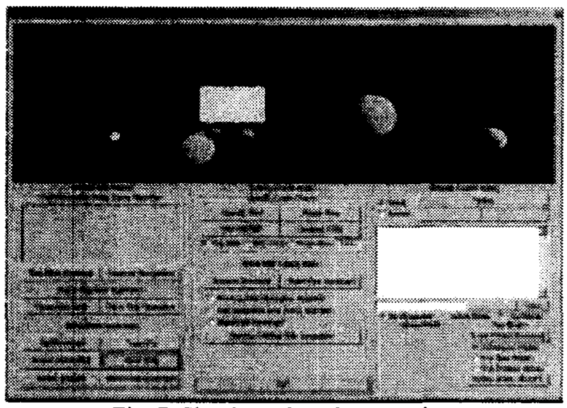


Fig. 7. Simulator GUI for our robot

In Fig. 7, upper block of figure shows the plain view of walking with obstacle, rightmost of lower blocks is an actual robot commander via serial channel. Using this block, we can send a robot actual angles that can move free walking. Middle block represents 3D control unit, which we can investigate the movement of robot with OpenGL engine, and confirm the validity of the desired robot operation. The leftmost part deals with kinematics and leg trajectory generation of robot after giving the parameters for the robot. We can use the optimal walking trajectory after verifying the given information from the above simulator. Fig. 8 shows path generation simulation result. In Fig. 8., obstacles are three, and circles around obstacles show boundary to walking without collision.

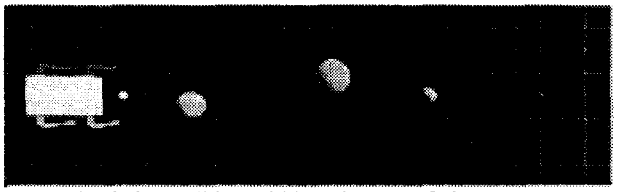


Fig. 8. 3D walking within existence of obstacles

And two small spheres mean starting point and end point respectively. Solid line that connects from starting point to end point is body path trajectory generated by fuzzy algorithm without collision against obstacles. From Fig. 9 to Fig. 11 show the angular acceleration of each joint at leg 1. Using the induced angles from genetic algorithm, we found that the sum of acceleration of shoulder's pitch joint reduces about 40%, shoulder's yaw improve about 26%, and knee's pitch improve about 13%.

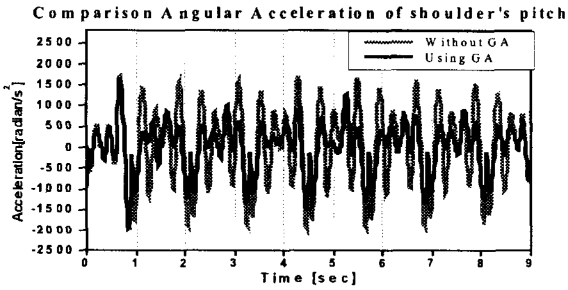



Fig. 9. Angular acceleration of leg 1 shoulder pitch

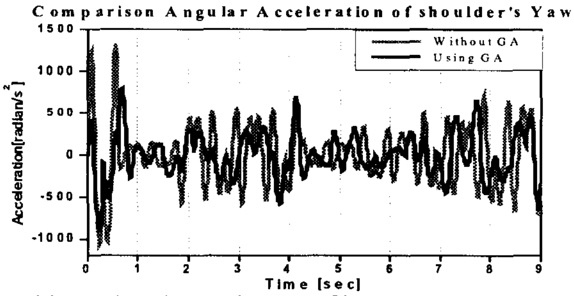



Fig. 10 Angular acceleration of leg 1 shoulder yaw

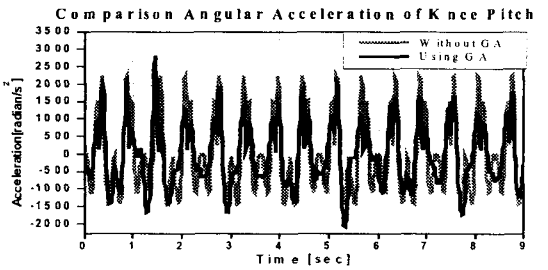



Fig. 11. Angular acceleration of leg 1 knee pitch

## VI. CONCLUSION

It is difficult to avoid the obstacle for the quadruped robot because avoidance path is related with leg's trajectory. Also to make leg's trajectory, robot should know the body drive trajectory considering with leg's movement. Therefore we need the relation algorithm between their trajectories. Other important point for designing a robot is to realize the smooth motion generation of legs, which is essential to implement the stable walking.

In this paper, we suggest genetic algorithm and fuzzy algorithm to solve those problems. We propose body path generation with fuzzy algorithm. From fuzzy algorithm, we could gain optimal path data that avoid obstacles. After making this path, we could convert legs via point to obey body path data. At same time, we apply the genetic algorithm help to gain optimal velocity and acceleration to walk smooth. Through these processes, robot can avoid obstacles with smooth walking. The suggested method is verified with our robot SERO-V.

In the future, we will implement vision system and various sensors to obtain the real data from external environment, so we can realize more accurate motions and approach the human interactive behavior between human and robot.

## REFERENCES


[1] Masahiro Fujita, "AIBO:Toward the Era of Digital Creatures," The International Journal of Robotics Research, Vol. 20, No. 10, pp781-794, 2001.

[2] G. S. Hornby, S. Takamura, J. Yokono, O. Hanagata, T. Yamamoto and M. Fujita, " Evolving Robust Gaits with AIBO," IEEE Robotics and Automation, Vol. 3, pp.3040-3045, 2000.

[3] P. Buschka, A. Saffiotti and Z. Wasik, "Fuzzy Lankmark-Based Localization for a Legged Robot," IEEE/RSJ International Conference, Vol. 2, pp1205-1210, 2000.

[4] Ryo Kurazume, K.Yoneda, and S. Hirose, "Feedforward and Feedback Dynamic Trot Gait Control for Quadruped Walking Vehicle",Autonomous Robot 12,157-172,2002.

[5] Xuedong Chen, Keigo Watanabe, Kazuo Kiguchi and Kiyotaka Izumi, "An ART-Based Fuzzy Controller for a Adaptive Navigation of a Quadruped Robot", IEEE/ASME Transactions on mechatronics, vol. 7, no. 3, pp.318-328, 2002.

[6] H.Zhung, J.Wu ,and W.Hung, "Optimal Planning of Robot Calibration Experiments by Genetic Algorithms," IEEE Conf. on Robotics and Automation vol.2, pp.981-986, 1996.

[7] K.Shimojima,N.Kubota,T.Fukuda,"Trajectory Planning of Reconifigurable Redundant Manipulator Using Virus- Evolutionary Genetic Algorithm," IEEE Conf. on Industrial Electronics, Control, and Instrumentation, vol.2, pp.836-841, 1996.

[8] K. H. Shim, B. H. Lee, and J.G. Kim, "Control Balancing Weight for IWR Robot by Genetic Algorithm", Proceeding of the 11th KACC, pp1185-1188, 1996.