35th Annual International Conference of the IEEE EMBS
Osaka, Japan, 3 - 7 July, 2013

Ratbot Automatic Navigation by Electrical Reward Stimulation based on Distance Measurement in Unknown Environments

Liqiang Gao[1] Chao Sun[2] Chen Zhang[3] Nenggan Zheng[4] Weidong Chen[5] Xiaoxiang Zheng[6]

***Abstract*— Traditional automatic navigation methods for bio-robots are constrained to configured environments and thus can't be applied to tasks in unknown environments. With no consideration of bio-robot's own innate living ability and treating bio-robots in the same way as mechanical robots, those methods neglect the intelligence behavior of animals. This paper proposes a novel ratbot automatic navigation method in unknown environments using only reward stimulation and *distance* measurement. By utilizing rat's habit of *thigmotaxis* and its reward-seeking behavior, this method is able to incorporate rat's intrinsic intelligence of obstacle avoidance and path searching into navigation. Experiment results show that this method works robustly and can successfully navigate the ratbot to a target in the unknown environment. This work might put a solid base for application of ratbots and also has significant implication of automatic navigation for other bio-robots as well.**

I. Introduction

Bio-robotics is a new research field which focuses on exploiting animal's locomotor ability and other unique aptitudes to serve human-beings through Brain Computer Interface (BCI) techniques. Electrical stimulation as control commands delivered into sensory cortex directly is used to manipulate animals to perform specified behaviors, from easily pressing a lever to finishing complex 3D terrain traversal. There have been different animals developed into bio-robots, such as rats [1], beetles [2], sharks [3], and geckos [4]. Bio-robots have significant advantages over mechanical rivals for their athletic agility, power supply, and concealability [5]. Of all the superiorities, the most essential one is that they possess great locomotive adaptability and cognitive capability in different environments, which makes them ideal tools for target searching in unknown environments.

However, the automatic navigation methods for bio-robots in unknown environment is still a challenge that impedes their practical applications. There are many traditional controlling methods for mechanical robots in unknown or dynamic environment, including the heuristic searching such as D* algorithm [6], behaviors based methods such as local field potential algorithm [7] and machine learning method like fuzzy learning [8] or reinforcement learning [9]. Those methods mainly focus on obstacle detection and avoidance. Subject to the complexities of various environments, no universal methods apply to arbitrary scenarios. In contrast, animals, e.g., rodents, can readily avoid obstacles, explore the surroundings and search for targets over different environments. This ability of spatial recognition benefits from animals' intelligence evolved through millions of years. The biological intelligence of animals would be the very solution to the problem in navigation methods. In bio-robots navigation, the controlling algorithm should *implant* the assigned destination into the animals' willing as their desired target, leaving the obstacles and collision avoidance to be handled by the animals themselves.

Previous studies implement the automatic navigation of bio-robots only in configured experimental maze [10][11] instead of the real world scenario, with animal intelligence totally ignored. Furthermore subject to the load capacity of the animals, the complicated and heavy sensors cannot be equipped on bio-robots' bodies. Meanwhile the movement of animals will cause these sensors to tremble drastically which affects sensor precision. Therefore the sensory data for automatic navigation in bio-robots should be carefully examined. In this work, we employ the *distance* between the bio-robot and the target as the only locomotion information for navigation. This parameter requires few sensors and little calculation to obtain. More importantly, the navigation based on *distance* has practical significance. This *distance* can be directly sensed by animals, reflected on their neural systems and eventually be decoded from the brains directly through BCI techniques. In future intelligence-hybrid bio-robots, the *distance* will be interpreted from the brain and generates controlling commands to guide bio-robot in navigation tasks as a close-loop of BCI system.

In this paper, we proposed a novel ratbot (bio-robot implemented by a living rat) automatic navigation method in unknown environments with electrical reward stimulation based on *distance* measurement. Our method focuses on giving/depriving the reward when the ratbot walks towards/away from target in navigation tasks, while leaving the problems such as how to avoid obstacles and how to choose a feasible route to ratbots themselves. The reward commands are decided and sent by evaluating aforementioned *distance* information. The results show that by taking advantage of the animal's biological intelligence, our method successfully implements the automatic navigation for ratbot in an unknown environmental maze with a simple controlling logic.

This work was supported by grants from the 973 Program (No.2013CB329506), the National Natural Science Foundation of China (No.61031002, 61003150), the Fundamental Research Funds for the Central Universities.

[1]Liqiang Gao, [2]Chao Sun, [3]Chen Zhang, [4]Nenggan Zheng, [5]Weidong Chen, [6]Xiaoxiang Zheng are with Qiushi Academy for Advanced Studies(QAAS), Zhejiang University, Hangzhou 310027, China. [4]Nenggan Zheng is the corresponding author (e-mail: qaas@zju.edu.cn).

[6]Xiaoxiang Zheng is also with the Department of Biomedical Engineering, Zhejiang University, Hangzhou 310027, China. [1]Liqiang Gao, [2]Chao Sun, [3]Chen Zhang are also with College of Computer Science, Zhejiang Univerisity, Hangzhou 310027, China.

II. METHODS

A. Ratbots

The electrical stimulation in Medial Forebrain Bundle (MFB) cortex will generate intensive excitement for rodents. In this way, we can make the rat act as a mobile robot in navigation experiments. Due to the reward-seeking instinct, the rat will keep walking forward for more rewards. We have built an automatic navigation system for ratbot based on electrical reward stimulation in previous work [12].

Rodents have excellent talents in environment exploration to search for food and water. Besides, rats show special behaviors such as preferring walking against vertical walls which is referred to as *thigmotaxis* terminologically [13]. The searching capability and roaming habit are the physiological basis of our navigation method.

B. Automatic Navigation Method

1) Principle: The principle of our navigation method is inspired by the classical BUG algorithm [14] in traditional robotics. The basic idea of BUG is to guide robots to: 1) advance along the line from start point to target point. 2) circle around any obstacles in the way until it returns to the original route. Combining the characteristics of ratbots and the rules of BUG algorithm, we define two basic controlling strategies in our navigation method: *Open_Control*(OC) and *Tunnel_Control*(TC).

OC strategy regards the environment as an open field where there is no obstacles blocking ratbot's way. When the *distance* information indicates that the ratbot is approaching towards target, the reward electrical stimulation is given immediately to induce it to move on, otherwise the reward is deprived. The ratbot will realize the wrong behaviors when rewards are cut off and try other routes based on its error-and-trial learning ability. In this way, the ratbot is guided towards the destination gradually as shown in Fig. 1(left).

However, when the ratbot walks into a dead end as in Fig. 1(right), OC strategy can't work. No matter which direction ratbot chooses to walk out of dead end, the *distance* gets larger and ratbot will not get reward to move further. Our method employs strategy TC to handle this ratbot trapped situation. In TC, the reward is given continuously to induce the ratbot to move regardless of whether it walks towards or away from target. Due to the *thigmotaxis* characteristic, under continuous reward stimulation, ratbot shall walk against the boundaries and thus circles around the rims of obstacles. Once the *distance* is less than the previous local minimal value, the strategy is switched back to OC.

There are two key issues that our method focuses on. Firstly, the ratbot trapped scenario should be detected by the *distance* parameter real-timely. Secondly, during TC controlling process, there are possibilities that the ratbot chooses a wrong direction which is target-unreachable. Our method should make the ratbot realize the error and try the other direction.

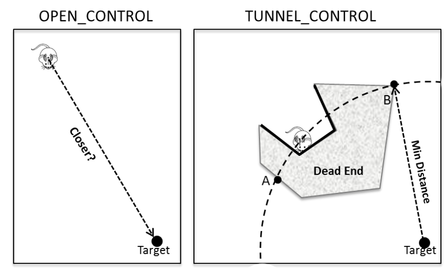

Fig. 1. Automatic Navigation Strategies. 1) In OC, ratbot only gets reward stimulation if it's getting closer to target, whereas in TC, continuous reward stimulation is given regardless of ratbot's movement. 2) When the system determines ratbot has walked out of dead end at point A or point B by getting nearer to target compared with the minimum *distance* in the dead end, the controlling strategy is switched back to OC. 3) The method judges when to alter navigation strategy from OC to TC and vice-versa utilizing only the *distance* to target.

2) Dead End Detection: In the dead end, ratbot will walk back and forth and the *distances* recorded during the period will swing much more frequently than normal. Feature point detection is used to detect abnormal *distance* swing in our navigation method. We account those points which are spike-shaped in the *distance* diagram as feature points. Explicitly Those satisfying

$$(Dcur - Dprior) * (Dcur - Dnext) > 0$$

are classified as feature points. Among the formula, $Dcur$ stands for the current distance. $Dprior$ and $Dnext$ are the chronologically prior and next distance respectively.

Dead end is detected in a *time-window* which correlates to a *distance* record interval. The length of a time-window is decided by *interest distance gap* which is a parameter regarding to the depth of dead ends. When feature points in the *time-window* exceeds a proper threshold, ratbot is defined as trapped in a dead end.

3) Back Trace Restriction: As mentioned previously, some dead ends only have one direction out. TC strategy can't navigate ratbot out if it heads into the wrong direction in these dead ends. To cope with this situation, the back trace range is restricted to a threshold. Whenever the difference between current *distance* and minimum *distance* in a dead end is more than the threshold, ratbot is temporarily believed to have walked to the wrong direction and the reward stimulation is deprived off to prevent ratbot from moving further from target. Continuous reward is resumed when ratbot falls into the back trace range threshold again and the method then will try to navigate ratbot to the opposite direction.

The back trace restriction threshold is not static but dynamically increasing so as to adjust to the actual depth of the current dead end. For each excess of back trace range threshold, the threshold auto-increments exponentially as the formula below:

$$T_{new} = T_{old} * f^N$$

T_{new} and T_{old} are the new and old threshold value respectively. The incremental factor is noted as f. N is the number of times that ratbot walks out of the threshold in a dead end. If a dead end is very large, TC shall restrict the back trace range to the current threshold. Ratbot will make several attempts to walk out of the dead end while the threshold increases to the depth of dead ends. If ratbot walks back and forth for too many trials and N exceeds a predefined value, we then conclude that ratbot can never walk out of the dead end.

C. Controlling Algorithm

The controlling algorithm for automatic navigation is as below.

```
Data: the distances between the ratbot and the target
Result: navigation SUCCESS or FAILURE
initialization;
while ratbot has not reached the target do
    read distance;
    if current controlling strategy is OC then
        control ratbot with OC;
        if ratbot is detected entering a dead end then
            alter controlling strategy to TC;
        end
    end
    if current controlling strategy is TC then
        control ratbot with TC;
        if ratbot is detected out of the dead end then
            alter controlling strategy to OC;
        end
    end
    if no time left or too many trials in a dead end then
        return navigation FAILURE;
    end
end
return navigation SUCCESS;
```

Algorithm 1: Controlling Algorithm

III. EXPERIMENTS AND RESULTS

A. Experiment Setting

We constructed an experimental environment with obstacles using rectangle-shaped wood baffles. The baffles were of a height of 20cm and could prevent ratbot from climbing over. They were put into a restricted rectangle planar region of 1.7m long and 1.4m wide, and were fixed at constant locations during experiments. Details of the environment formation can be found in Fig. 3. We also had a camera overhead to capture the movement of ratbot. The camera was solely used to obtain the distance of ratbot to target with computer vision techniques and all the environment information from the camera was not used for navigation.

In order to evaluate the navigation results, we initially used manual control to set up evaluation standards. Different trials were conducted in an open field or an environment with obstacles, and the time consumption of each navigation trial was recorded. For automatic navigation in the environment with obstacles, the success threshold was set to twice the average time consumption of manual navigation. The average time of manual navigation was 45.5s. So if the time consumption was more than 91.0s, the automatic navigation trial was judged as a failure trial.

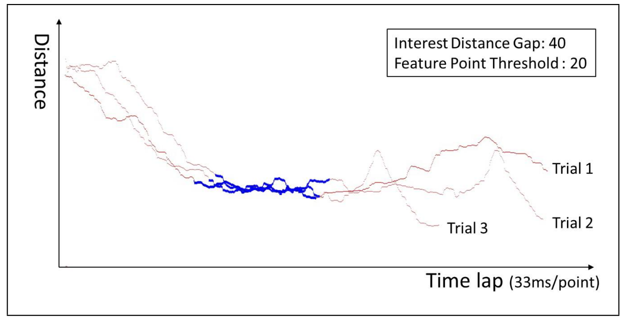

Fig. 2. Dead End Detect Process. The bold line segments in the diagram show that our algorithm uses the *distance* in this interval and detected that ratbot had entered a dead end.

B. Navigation Results

We firstly did experiments to test the efficiency of dead end detection. Three independent trials were conducted with manual control following the rules of our new navigation method. Offline analysis of the video showed that feature point detection could correctly detect dead ends in real time. The dead end detection process is presented on Fig. 2.

The detailed results of the average time consumption are listed in table I. According to our navigation success standard, for 20 trials in our experiments, the success rate was 85%.

TABLE I

AVERAGE TIME CONSUMPTION OF NAVIGATION

Environment	No obstacle		With Obstacle		
Method	Manual	Auto	Manual	Manual & PP*	Auto
Time	14.3	18.5	45.5	23.9	73.7

*PP: Path Planning which is not in our method but added for comparison.

The typical locomotion trajectory of a ratbot in environment with obstacles is presented in Fig. 3. In the graph, the red dots on the curve indicate that the ratbot is given a reward stimulation under OC strategy, while the blue rectangle means that the ratbot is given a reward stimulation under TC strategy. The automatic navigation process starts when the ratbot walks out of the initiate region at point B. Both of point A and point C represent that the ratbot has been detected entering a dead end zone. When the ratbot passes point D, the distance to target becomes less than the minimal distance recorded in the dead end and thus controlling strategy is switched back to OC. Ratbot makes a sharp turn at point E because no reward stimulation is given.

In our experiment, there was one dead end having only one direction out and the other direction led ratbot back to its start place. The navigation process out of the dead end is shown in Fig. 4.

We observed that occasional false dead end detection occurred during automatic navigation. When a false dead end detection occurs, if ratbot is heading closer to target, then immediately the controlling strategy is switched back to OC.

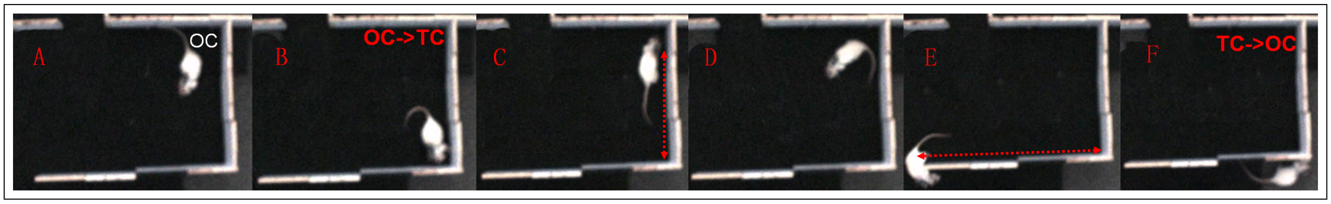

Fig. 4. Back Trace Restriction. The current back trace threshold is drawn as the red dotted line as in C and E. The ratbot is controlled under OC strategy in A. In B, it is detected to be trapped in a dead end and thus controlling strategy is switched to TC. The ratbot walks out of the threshold in C and reward is deprived off. The ratbot realizes that it has walked to the wrong direction and makes a turn in D. The threshold is incremented as in E, then the ratbot is navigated out of the dead end in F and navigation strategy is switched back to OC.

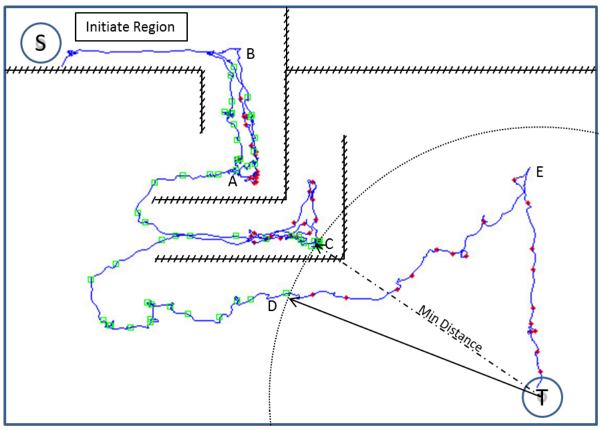

Fig. 3. Ratbot Navigation Trajectory. This graph represents the actual environment setting and the trajectory of the ratbot. The palisade shaped line stands for the barriers we arranged and the irregular curve in the middle is the ratbot's walking path. The start point is at the upperleft corner, noted as circle S, and the target is at the lowerright corner, noted as circle T.

And if ratbot is heading away from target, then according to the rules of our method, continuous reward stimulation is given to drive ratbot out of the false dead end. When the *distance* has exceeded the back trace range threshold, the reward stimulation is deprived. Ratbot shall turn around to get reward and move towards target again. So in a tolerable time span, a small amount of false dead end detection won't have much influence on the final navigation results.

IV. CONCLUSION AND FUTURE WORK

This paper has proposed a novel ratbot automatic navigation method in unknown environments. By taking ratbot's intrinsic ability of obstacle avoidance and path searching into consideration, this method integrates the intelligent behavior of ratbot and builds up a feasible solution for ratbot navigation in unknown environment.

In real navigation tasks like hazardous environment searching and rescuing, a lot of unexpected surroundings and objects will emerge. The influence they will have on the navigation of ratbot and ways to avoid them should be addressed before deploying ratbot into specific tasks. More work including getting the distance information from carried sensors or directly from the neural systems of ratbots needs to be done in the upcoming research.

ACKNOWLEDGMENT

The authors would thank Yan Cao for the rat surgery and Chaonan Yu for helping training rats.

REFERENCES

[1] S. Talwar, S. Xu, E. Hawley, S. Weiss, K. Moxon, and J. Chapin, "Behavioural neuroscience: Rat navigation guided by remote control," *Nature*, vol. 417, no. 6884, pp. 37–38, 2002.

[2] H. Sato, C. Berry, Y. Peeri, E. Baghoomian, B. Casey, G. Lavella, J. VandenBrooks, J. Harrison, and M. Maharbiz, "Remote radio control of insect flight," *Frontiers in integrative neuroscience*, vol. 3, 2009.

[3] S. Brown *et al.*, "Stealth sharks to patrol the high seas," *New scientist*, vol. 189, no. 2541, 2006.

[4] H. Li, Z. Dai, H. Tan, C. GUO, and J. SUN, "A remote system for gecko animalrobot," *Computer Technology and Development*, vol. 18, no. 8, pp. 16–19, 2008.

[5] C. GUO, Z. DAI, and J. SUN, "Current status and prospect of research on bio-robots," *Robot*, vol. 2, pp. 187–192, 2005.

[6] A. Stentz, "Optimal and efficient path planning for unknown and dynamic environments," tech. rep., DTIC Document, 1993.

[7] R. Arkin, "Motor schema based navigation for a mobile robot: An approach to programming by behavior," in *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, vol. 4, pp. 264–271, IEEE, 1987.

[8] H. Hagras, F. Doctor, V. Callaghan, and A. Lopez, "An incremental adaptive life long learning approach for type-2 fuzzy embedded agents in ambient intelligent environments," *Fuzzy Systems, IEEE Transactions on*, vol. 15, no. 1, pp. 41–55, 2007.

[9] H. Beom and H. Cho, "A sensor-based navigation for a mobile robot using fuzzy logic and reinforcement learning," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 25, no. 3, pp. 464–477, 1995.

[10] Y. Zhang, C. Sun, N. Zheng, S. Zhang, J. Lin, W. Chen, and X. Zheng, "An automatic control system for ratbot navigation," in *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*, pp. 895–900, IEEE, 2010.

[11] X. Zhang, C. Sun, N. Zheng, W. Chen, and X. Zheng, "Motion states extraction with optical flow for rat-robot automatic navigation," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pp. 976–979, IEEE, 2012.

[12] C. Sun, X. Zhang, N. Zheng, W. Chen, and X. Zheng, "Bio-robots automatic navigation with electrical reward stimulation," in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pp. 348–351, IEEE, 2012.

[13] I. Whishaw and B. Kolb, *The Behavior of the Laboratory Rat:A Handbook with Tests: A Handbook with Tests*. Oxford University Press, USA, 2004.

[14] V. Lumelsky and A. Stepanov, "Path-planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape," *Algorithmica*, vol. 2, no. 1, pp. 403–430, 1987.