Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2013, 5(9):529-535

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

Aggregation and docking strategies for mobile self-reconfigurable robots

Ming Zhong, Y.F. Yao, B. Huang

State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin, P. R. China

---

ABSTRACT

The reversible process between distributed robots and connected robot is the key issue of mobile self-reconfigurable robots, which can be divided into two phases of aggregation and docking. In this paper, corresponding strategies are proposed for robots enhanced with mobility. In the dynamic aggregation mechanisms, the optimal locations are determined based on the multi-object optimization of time cost, energy consumption, and system balance. Besides, a predictive chasing method is developed for collective robots to adjust position and attitude. Based on the aggregation results, docking strategy makes all the adjacent robots connect together via docking devices including MEMS-navigation module and embedded vision. All the strategies are applied to the control of mobile self-reconfigurable robots, and experiments are performed. The results show that the proposed strategies are effective for the implementation of reversible process.

Key words: Aggregation, Docking, Self-reconfigurable robots, Mobile robots, Multi-agent system.

INTRODUCTION

The concept of mobile self-reconfigurable (MSR) robots which are capable of implementing tasks as many individual modules or as a combined robot has been inspired from the common advantages of distributed robotics and self-reconfigurable robots. The key problem related to the design and implementation of such systems is how to achieve the reversible process between dispersing mobile robots and connected robot. Since the late 1980s, a large body of research has already shown that the component modules of MSR robotic system can be set up to address a variety of tasks under different environmental circumstances. Fukuda and Nakagawa proposed the first MSR robot CEBOT consisting of separable autonomous units, which was able to adapt itself to changing environments via dynamical reconfiguration among these units [1]. Motomura *et al*. investigated a concept SMC for planetary exploration, and described potential benefits of such systems in the context of autonomous all-terrain locomotion [2]. Brown *et al*. designed a prototype Millibot Trains composed of tracked mobile units, which allowed the units to engage/disengage under computer control, and the manually configurable train demonstrated the ability to climb stairs and vertical steps [3]. Mondada, O'Grady, and Gro *et al.* proposed a robotic concept SWARM-BOT in which the collective interaction was exploited by the swarm intelligence mechanism, and mechanical functionalities on the single robot allowed to address complex mobile robotics problems [4 - 6]. The afore-mentioned research extended the applicable view of MSR robots; however, most of the docking mechanisms were based on the weak mobility of individual modules, and very little research has been conducted into how modules with strong movement capacity can autonomously arrange themselves into configurations.

In this paper, we address strategies for the reversible process between distributed mobile robots and connected robot, which is the key problem of MSR robots. When the single robot is endowed with strong mobility, especially when the working radius of robots becomes large, localization and communication of robotic system become challenging. Hence we divide the reversible process into two segments, aggregation mechanisms that dispersing robots collect at the same location according to task requirement, and docking strategies that single robots connect each other one by one into a combined robot. After implementation of aggregation and docking, distributed robots transfer to self-reconfigurable robots, the latter is able to change its configuration, such as snake and ring shape, and adapt to tough terrains including ditches, humps, and slopes.

This paper is organized as follows. Firstly, we briefly present the Tanbot robotic platform that we use in the experimentation. The next part introduces detailedly the aggregation mechanisms and docking strategy respectively. The end part depicts the experimental set-up and the results, and some conclusions are provided.

THE TANBOT ROBOTIC PLATFORM

For our experiments, we use the improved robotic platform proposed before [7]. It is made up of multiple mobile autonomous robots called Tanbot (see Fig. 1) which can change the structure by extending its length to overcome small obstacles, and two or more Tanbots can connect each other to be an entity via physical docking mechanism.

Tanbot with dimension of 24 cm (L) × 22 cm (W) × 7 cm (H) is covered by two caterpillars, which are driven by two DC motors respectively to achieve straight moving and spot turning, and its maximum speed is around 0.8 m/s. The lift mechanisms fixed on the body frame are the essential parts for Tanbot, and one is used to lift the docking pin, the other is applied to rotate one caterpillar structure with respect to the other around the rotary shaft. Most of the electrical units are sealed in the body frame for protection, including central controller, navigation module, image processor, and motor driver. In particular, a miniature camera with 45 degrees view range is embedded in the hollow docking pin for guiding and monitoring.

Physical connections between Tanbot are established by the matching between docking pin and docking hole. After expanding the structure, each Tanbot can achieve connection with others via inserting its docking pin into another's docking hole. During this docking process, the MEMS-based navigation module measures the relative position and head direction to execute rough adjustment. Based on the result, the camera takes task over to precisely guide the active Tanbot to match the positive one.

DYNAMIC AGGREGATION MECHANISMS

In this section, we describe the dynamic aggregation mechanisms for distributed robots to collect in a small area.

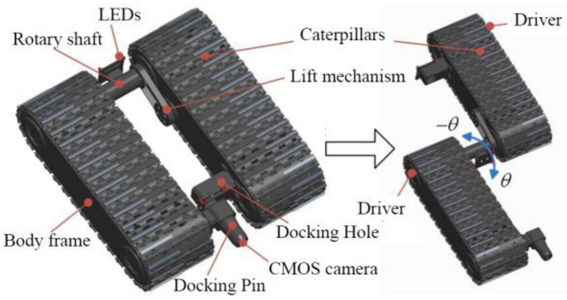

Fig. 1. Tanbot structure and self-reconfiguration by extending its length

Aggregation is the preparing work for docking, and its final results affect the docking efficiency. Due to the characteristic of docking mechanism, Tanbot has to change its structure first so as to reveal the pin and hole, and this particularity requires robots arranged as linearly as possible.

Before aggregation, the common destination should be determined by Tanbots which participate in the process. There are two ways to choose the aggregation locations, static and dynamic. Static locations are computed based on the positions of all robots at the start of aggregation, which is easy and unchangeable. However, the ever optimal locations would change along with the movement of robots. Thus we choose the dynamic way that all robots compute the optimal locations every time slot based on their current positions.

Optimization Conditions of Aggregation: All robots are not identical due to manufacture, and each robot has different conditions of speed and energy. To make all Tanbots arrive around approximately same time and avoid member lost during aggregation, we take several aspects including time cost, energy consumption, system balance, and robot characteristic into consideration to optimize the dynamic locations.

Firstly, the aggregation time cost T_s depends on the robot which arrives at location last, and too early arriving of some robots is no help to reduce T_s. Secondly, in view of the robotic system endurance, aggregation process should consume energy as less as possible, so it is necessary to calculate the system energy consumption, which is proportional to the total distances S of all robots. Finally, in general, the distances s_i that robots need move to their locations are not same, which leads to different energy consumptions. Measures should be taken for robot with less remaining energy q_i to prevent it from exhaustion. Thus we can establish the optimization conditions of aggregation as

$$\begin{cases} T_s = \max\{s_i / v_i\} & \rightarrow \min \\ S = \sum_{i=1}^{n} s_i & \rightarrow \min \\ Q_i = q_i - F_i(v_i) \cdot s_i & \rightarrow \max \end{cases} \quad (1)$$

Optimization of Dynamic Locations: The ideal aggregation result is that all Tanbots are located in a straight line for follow-up docking. We adopt least squares method to get the dynamic straight line, which attains minimum sum of distances for robots moving to the destinations in the target line.

Set a constant spacing interval Δ_d between near Tanbots, if one location is settled, then the rest of other locations are determined based on the required straight line. Therefore, the problem of calculating optimal dynamic locations is transferred to the multi-objective optimization of one location in the fitting line. Based on Eq. (1), we can derive the object function as

$$\begin{cases} f_1(x_d, y_d) = \max\left\{s_i(x_d, y_d) / v_i\right\} \\ f_2(x_d, y_d) = \sum_{i=1}^{n} s_i(x_d, y_d) \\ f_3(x_d, y_d) = q_i - F_i(v_i) \cdot s_i(x_d, y_d) \end{cases} \quad (2)$$

where (x_d, y_d) is the coordinate of one endpoint in line (i.e. the first or end robot in the linear team); s_i is the distance of robot i would move to its destination, and v_i is its speed; q_i is the remaining energy of robot i at each calculation; $F_i(v_i)$ is the resistance function in inverse proportion to robot speed v_i.

Combining Eq. (1) and (2), the multi-object optimization model can be established, and we adopt method of targets Multiplication and division to transfer it as

$$\min_{(x_d, y_d) \in L_d} \frac{f_1(x_d, y_d) f_2(x_d, y_d)}{f_3(x_d, y_d)} \quad (3)$$

All locations are restricted by the target line L_d, so Eq. (3) can be solved by one-dimensional search technique, and then the optimal dynamic locations of all Tanbots are determined.

Adjustment of Position and Attitude: After all Tanbots arriving at their destinations with different heading direction (see Fig. 2), they disperse on both sides of the fitting line because of navigation errors. For conveniently docking, adjustment of position and attitude is necessary.

We proposed a predictive chasing method to simultaneously adjust the positions and attitudes of all Tanbots in the belt area with width of Δ_w. As shown in Fig. 3, set a virtual location O_1 in the end part of fitting line, and the close robot A_1 moves to O_1, then the robot A_2 which is near A_1 moves to the midpoint between *A1* and O_1, and the robot A_3 which is near A_2 moves to the midpoint between A_1 and A_2, and so on.

The choice of the virtual location O_1 should take adjustment efficiency and robots collision into consideration, and we set it with a constant distance from the close Tanbot A_1. In each time interval Δ_d, robot i moves a distance Δ_{Si} to a new position A_i', then it recalculates the new midpoint and adjusts its attitude to move. After several time intervals, all Tanbots move closer to the target line, and their directions gradually become same, which has been proved truly by geometry.

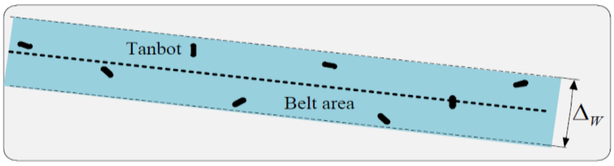

Fig. 2. Illustration of the aggregation results, all Tanbots disperse on both sides of the fitting line with different attitudes

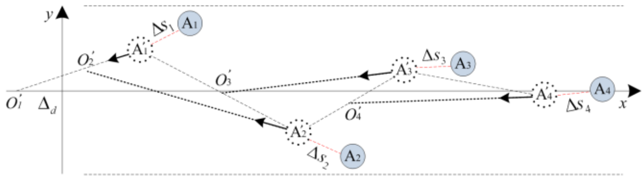

Fig. 3. Illustration of the predictive chasing method, Tanbots in the belt area chases aims for position and attitude adjustment

DOCKING STRATEGY

In this section, we introduce the docking strategy for the collective robots connecting each other to be a combined robot. Due to the roughness of terrain and errors of navigation modules, there is still a small relative angle α between two adjacent robots (see Fig. 4). While docking, one keeps still (the passive), and the other adjusts its attitude (the active).

Tanbot is endowed with a MEMS-based navigation module which measures position and two-dimensional attitudes, so it can roughly compare the relative positions and angles between two robots. Specially, there are four red LED lights fix in the rear of Tanbot, and another four are fixed around the docking hole. Camera takes photos of the LEDs array, and then compares two side length of the square (L_h and L_v) to calculate the relative angle equivalently (see Fig. 5), which makes Tanbot able to get precise estimation of attitude.

Fig. 6 shows the flow chart of docking process. Firstly, the active robot gets the rough relative angle α via the MEMS-based navigation module, and then adjusts its attitude to reduce it. At the same time, the CMOS camera scans the front; once the four red LEDs are within its field of view, the camera takes guiding work over. The active robot gets precise relative angle α via camera, and adjust its attitude till α is smaller than the threshold value α_t, then the two adjacent robots expand their structures, and the active robot approaches to insert its pin into the hole of the passive robot (see Fig. 7).

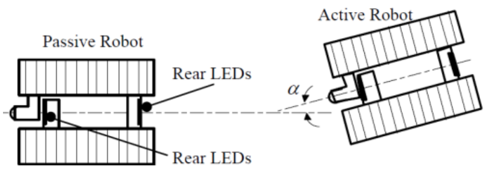

Fig. 4. Small relative angle between two adjacent robots

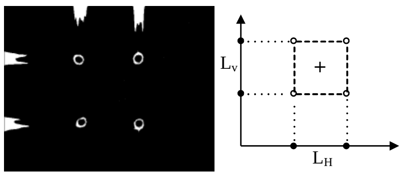

Fig. 5. Image processing of the square array red LEDs; the left is a binary image by filtering, and the right is the projection of light spots

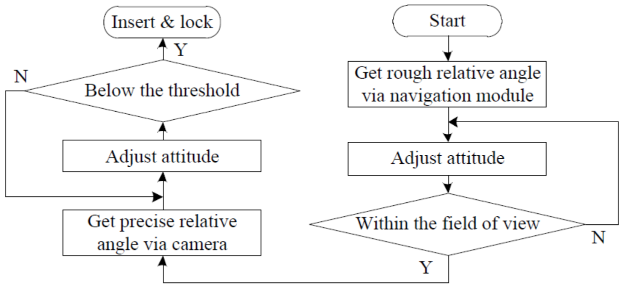

Fig. 6. Flow chart of the docking process between two adjacent robots

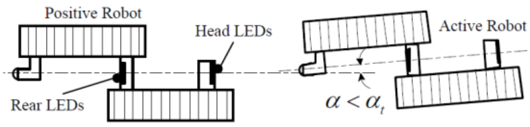

Fig. 7. After vision guidance, if the relative angle $\alpha<\alpha_t$, the active robot approaches to docking.

EXPERIMENTS OF AGGREGATION & DOCKING

Experimental Set Up: To achieve information exchange among Tanbots, we adopted centralized control based on Multi-agent System (MAS). And the Blackboard Model was established for information disclosure of all system agents including Tanbots, monitoring agent, interaction agent, conflict resolution agent, aggregation agent, adjustment agent, and docking agent, as shown in Fig. 8.

Therefore, all Tanbots can visit Blackboard via wireless communication with host computer, and get any expected information, such as other's coordinate, destination, speed, energy, and so on, which is propitious to realize the reversible process from distributed mobile robots to a connected robot.

Docking Based on Vision: To test docking efficiency between adjacent robots only with vision guiding, two Tanbots are placed with spacing L and relative angle α (see Fig. 9).

Because of image clarity, the spacing *L* is set smaller than 68 cm, and the threshold value of relative angle is set 4 degrees based on the tolerance of docking mechanism. Experiments were implemented (see Fig. 10) by increasing the relative angle α to test the docking time (see Table 1.). It is obvious that the process time grows rapidly as *α* increases, and the best range of *α* for docking only with vision guidance is (0 degrees, 15 degrees].

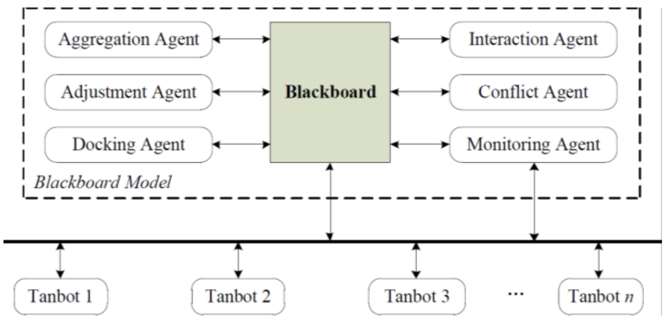

Fig. 8. Architecture of centralized control based on MAS and Blackboard.

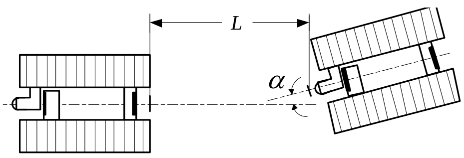

Fig. 9. Experimental set up between docking between adjacent Tanbots

Table 1. Process time of docking with different relative angles

α (degree)	3	6	9	12	15	18
t (second)	2.3	7.5	8.8	10.9	12.1	23.4

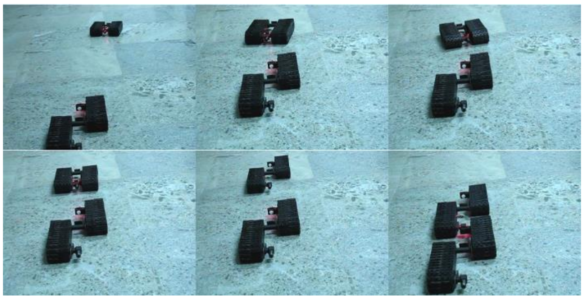

Fig. 10. Docking process between adjacent robots based on vision guiding

Aggregation and Docking of Multiple Tanbots: To verify the aggregation and docking strategy, four Tanbots are randomly placed within a square of 6 m x 6 m (see Fig. 11). Each robot periodically uploads its status information to the Blackboard of host computer, where it can also get others' information. Cartesian coordinate system was established according to the square area, and the coordinates of distributed Tanbots were marked and published in the Blackboard of the host computer, which makes all robots able to the positions of teammates.

Completion of the transition process from four distributed Tanbot to a combined robot cost 81 seconds, and the experimental results are shown in Fig. 12. Based on the discrete coordinates recorded by Blackboard of host computer, tracks of all robots can be drawn; and also the expected traces can be simulated by MATLAB according to the initial coordinates of robots (see Fig. 13).

Comparing the real tracks and simulated traces, we can find that the final formations do not coincident. There are several factors lead to this error, such as sensor accuracy, terrain roughness, and navigation errors. Luckily, the reversible process between distributed robots and connected robot requires low accuracy for the final position of the combined robot, which does not affect the practical application of mobile self-reconfigurable robots.

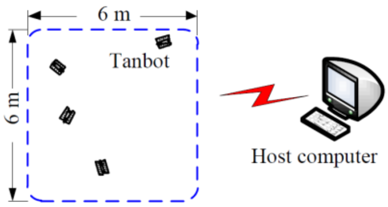

Fig. 11. Experimental set up of aggregation and docking among multi-robots

Fig. 12. Process of aggregation and docking among multi-robots

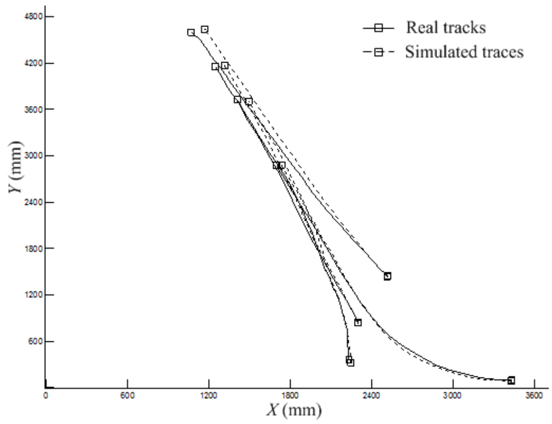

Fig. 13. Comparison between real tracks and simulated traces

CONCLUSION

Mobile self-reconfigurable robots can execute the reversible process to realize the transfer from distributed robots to a connected robot. In this study, we divide the reversible process into two parts, aggregation and docking. The aggregation mechanism determines the optimal location for all robots to collect and then adjusts their positions and attitudes preparing for docking. The docking strategy makes the collective robot connect each other one by one to be an entity. Experiments including docking between adjacent robots and overall aggregation & docking among multiple robots are performed to verify the proposed strategies.

Due to some objective factors, there are slightly differences between experimental results and simulated traces. However, this error does not affect the practical application of mobile self-reconfigurable robots. One point should be noted that we adopt centralized control method based on Blackboard, which is not suitable for a large quantity of robots to implement the reversible process. Nevertheless, taking the characteristic of Tanbot into consideration, too many such robots collect and connect together is not necessary. Therefore, the strategies proposed in this paper are feasible and efficient for MSR robots made up of a certain number of members.

Acknowledgments
This work was supported by the National Natural Science Foundation of China (Grant No. 61203359), and State Key Laboratory of Robotics and System (HIT) (Grant No. SKLRS-2012-ZD-05).

REFERENCES

[1] HB Brown Jr; JM Vande Weghe; CA Bererton; PK Khosla. *IEEE ASME Trans Mechatron*, **2002**, 7: 452-461.
[2] T Fukuda; S Nakagawa. *IEEE International Conference on Robotics and Automation*. Philadelphia, USA, **1988**. IEEE, pp: 1581-1586.
[3] R Gro; M Bonani; F Mondada; M Dorigo. *IEEE Trans. Rob.*, **2006**, 22: 1115-1130.
[4] K Motomura; A Kawakami; S Hirose. *IEEE International Conference on Robotics and Automation*. Taipei, Taiwan, September, **2003**, Institute of Electrical and Electronics Engineers Inc., pp: 63-68.
[5] F Mondada; GC Pettinaro; A Guignard. *IEEE Rob Autom Mag*, **2005**, 12: 21-28.
[6] R O'Grady; R Gro; A Christensen; M Dorigo. *Autonomous Robots*, **2010**, 28: 439-455.
[7] M Zhong; W Guo; JA Xu; LN Sun. *High Technol Letters*, **2009**, 15: 26-31.