



Journal of Applied Sciences

ISSN 1812-5654

science
alert

ANSI*net*
an open access publisher
<http://ansinet.com>

Moving Chain: A Surround-and-Push Method in Object Pushing Using Swarm of Robots

A. Khozaee and A. Ghaffari

Faculty of Mechanical Engineering, K.N Toosi University of Technology, Tehran, Iran

Abstract: In this research, a new approach to develop an object pushing system with a swarm of mobile robots is introduced. This is an attempt to design and develop a system for a swarm of robots and their behaviors to surround and push an object. The robot behaviors are devised in such a way that the robots would easily develop a swarm formation as an orbit around the object and push it toward the desired goal. Roughly speaking, a rotating orbit of robots around the object is developed first. The object is moved toward the goal as the orbit moves to the desired goal configuration. Based on this idea, using a simple decision making, each robot can easily determine its proper action to push the object while remaining in the robot formation. Fuzzy controllers are used to develop behaviors in each robot. Each robot makes decision and behaves individually without any direct communication with others. Based on this individually behavior the group cooperation will perform. The main characteristics of the developed system are independency of the method to the object kinematics and its shape. Moreover, as a result of a swarm system, the method is not dependent on the number of robots in the system. Simulation results are given to support the proposed approach.

Key words: Swarm robotics, stigmergy, object manipulation, quasi-static object manipulation, fuzzy controller

INTRODUCTION

Multi-agent systems and artificial intelligence communities are interested in using a swarm of robots to achieve some tasks. Understanding nature such as social insects, birds and fish (Fig. 1) helps designing swarm systems in robotics and intelligent systems (Sharkey, 2006). Artificial intelligence utilizes swarm robotics to build systems consisting of a large number of autonomous robots. Very recent studies inspired by ethological studies showed that a swarm of simple agents is capable of feats far greater than what each single unit is able to achieve alone (Pettinaro *et al.*, 2005). Using a huge number of robots assists in achieving a specific task without a need to a centralized control or an explicit communication (Chaimowicz *et al.*, 2004).



Fig. 1: A group of fish moving in a swarm formation

The idea of decentralized control is essential. In swarm systems no leader exists. The organization is completely flat, there is no hierarchy and all the agents in the system have the same amount of influence (Bjerknes *et al.*, 2007). Furthermore, robots in a swarm robotic system typically can not directly communicate with all other robots. Rather, each robot can only communicate directly with a few of its nearest neighbors (local communication) or indirectly by means of stigmergy. Stigmergy communication, an indirect one, is a way that robots communicate through the environment, as ants use pheromone to change the environment and notify others (Bonabeau *et al.*, 1999). In other words, by means of physically modifying the environment they are in, others can understand the changes happened to the environment. Each unit simply detects the state of its environment and reacts accordingly with primitive actions supposedly known by the whole group (Chaimowicz *et al.*, 2004).

Making use of simple and primitive actions in such systems directs to a simpler approach compared to other multi-agent systems (Martinoli *et al.*, 2004). These characteristics make swarm robotic suitable for tasks requiring robustness, flexibility and scalability. In addition these systems can scale up and down in time, providing redundancy in performing the task (Sahin, 2004).

It is very robust to hardware and software failures, because execution of a desirable task does not depend on any specific unit. A task can always be carried out by the

group regardless of its size and of the initial distribution of its members, so it is reliable and flexible. As the system is not dependent on any specific unit and also the number of robots can get more or less, it is scalable. In fact the system design with less monitoring is a result of such characteristics.

Whereas, swarm robotic can cope with these features, some studies are done by use of this concept such as searching (Haverinen *et al.*, 2005) and foraging (Steele and Geb, 2007). In addition to these common applications, it is appealing to exert this technique on many applications of multi-agent systems.

One of the robotic topics which has attracted multi-agent and robotic communities is object manipulation. Many researches have been interested in pushing the object using a small group of agents to achieve the task (Bernheisel and Lynch, 2005; Ahmadabadi and Nakano, 2001). Most of these methods are dependent on the number of robots in the team. Besides, earlier study have a complicated control algorithm and coordination method in the team. Furthermore, involving larger amount of robots to conquer the object manipulation task leads to a more fault tolerant system but with a more complexity drawback in the cooperation and coordination methods. Simplicity of swarm robotic techniques can assist to handle these problems of complexity.

Several techniques were introduced in order to reduce the complexity in object manipulation systems; Taking advantage of task assignment by Ahmadabadi and Nakano (2001) provided a fault-tolerant system with a simple algorithm where all the complexities were resolved by the use of a proper mechanics. In Aminaiee and Ahmadabadi (2007) a learning approach to the object pushing was introduced to overcome the complexity in designing cooperation and coordination protocols between agents. It is involved with substantial complication of a learning system. A caging algorithm is proposed by Wang *et al.* (2005) to entrap the object in a group of robots formation and move the object while the team of robots moves toward the goal maintaining their formation. Wang *et al.* (2005) a leader-follower type formation control algorithm were introduced to realize object transportation tasks. Where, in leader-follower systems, one of the agents in the team presents the central role as a leader and other agents in the team behave based on the state of the leader in the system. A similar way to caging method is to enclose the object using rope and pulling the rope by robots (Maneewarn and Detudom, 2005; Donald *et al.*, 2000).

Multi-agent object manipulation systems are classified as centralized or decentralized systems with

respect to cooperation and coordination methods between agents in the team. In a centralized control system, huge computational power and communication capacity is required in the controller depending on the number of agents in the system (Yang *et al.*, 2003). Additionally, any failure in the host will result in failure in the entire system. An example to these kinds of systems (Wang *et al.*, 1994), where an object pushing system consisting a host and a group of distributed vehicle robots is introduced. Leader-follower systems are considered as intermediate methods between centralized and decentralized systems; (Yang *et al.*, 2003; Hirata *et al.*, 2000). In another point of view several studies have been done on object manipulation with mobile robots equipped with grippers to make use of pushing and pulling mechanism to manipulate the object (Groß *et al.*, 2006; Hirata *et al.*, 2000; Wang *et al.*, 1994). In these kinds of approaches, mechanics of the grippers in the robots play an important role in the system performance while restricting the ability of the system to manipulate object regarding its size and weight. Meanwhile, in these methods any individual robot has a single constant point with the object during manipulation which may decrease robots utilization.

Some contributions of swarm robotics to object manipulation include Pettinaro *et al.* (2005), Groß *et al.* (2006), Zhang *et al.* (2007) and Stilwell and Bay (1993). Although Stilwell and Bay (1993) had robots to act like a swarm of ants but it utilized a robot as a leader to avoid collision during manipulation. In Groß *et al.* (2006) a swarm approach to object manipulation is introduced where six robots can pull a circular object by self-assembling and transporting it towards a goal in a self-organized manner. Another study performed by these robots, called s-bots, is introduced in Mondada *et al.* (2005) that cause transporting a child in a structured manner. Each robot has nine degrees of freedom and pulls the object by its gripper during transferring. Several studies, have been done on object manipulation with mobile robots equipped with grippers to make use of pushing and pulling mechanism to manipulate the object (Groß *et al.*, 2006; Hirata *et al.*, 2000; Wang *et al.*, 1994). These are not swarm-based manipulations but they have a common restriction; mechanics of the grippers in the robots play an important role in the system performance while restricting the ability of the system to manipulate object regarding its size and weight. Meanwhile, in these methods any individual robot has a single constant point with the object during manipulation which may decrease robots utilization. So, the systems based on pushing the object are focused here. Where, there is no need to have robots with grippers to move the object.

Zhang *et al.* (2007) introduced a swarm approach to object pushing and focused on task assignment in unknown environment. A swarm-based object manipulation is introduced by Bishop (2008), has some robots to push the object, but it is a centralized one. Object pushing by caging technique using a swarm of robots is also discussed by Michael *et al.* (2008) and Fink *et al.* (2007), where robots are able to control position of the object without any controllability on the object orientation. Three controllers were used to help the system in approaching, surrounding and transferring behaviors.

In this study a swarm of robots are to surround the object and push it toward the goal. The proposed method is independent from the number of robots available in the system. Task can scale up or down by increase or decrease in size of the object.

In multi-agent systems, those in which the robots are less dependent on the complicated sensors and explicit communication are preferable (Donald, 1995). Attaining this simplicity, robots are very simple in design and they do not acquire any communication with each other. For easy analyzing, because of contacting the robots to the object at any orientation, the circular robots are suitable. So, we applied some circular robots that have variant contact point with object and they have two degrees of freedom. These characteristics make units of system behave very simple, in other words they can push the object just by rubbing themselves to it. These simple robots are working in a group as a swarm while they apply an algorithm individually and their coordination lead to execution of an assigned task, object pushing towards the defined goal. They use stigmergy communication to know what changes are happened by other robots. The medium environment that is considered by robots as a stigmergy communication tool is the object.

A simple fuzzy inference system is used by each robot to provide the best action by each individual. Because of the fuzzy controller low level efforts, there is no need to apply a planner. Emergence of approach and surround behaviors by the use of the pushing and following behaviors is another attribute of the proposed method. Simulation results show that the object can be moved on any desired trajectory without any requirement to object information such as its shape and its center of friction.

PROBLEM DESCRIPTION

In this model a swarm of robots are to push a two dimensional object on a frictional surface. Assigning a swarm of robots to surround the object at all edges of the

object enables the system to apply force and torque in any direction. This permits the object to be pushed on any desired trajectory. Independency of the method to the object shape and size is an attribute required to be realized. It is also desired that no dependency on any unit to be existed in the system. The system must not be dependent on the number of robots. It is idealistic that agents can not communicate explicitly and just use stigmergy communication.

A group of disc shaped nonholonomic mobile robots are exploited for the object pushing task. Using nonholonomic mobile robots makes it difficult to push the object while following the object movements in all directions on its path. Hence, a proper pushing and following behaviors have to be designed to cope with this problem. Moreover, as robots may push the object against each other, pushing behavior has to be designed in a way to prevent conflicts and possible lock in the system.

The system must be able to move different objects of any shape in any arbitrary trajectory.

SYSTEM MODEL AND ASSUMPTIONS

It is assumed that the object is two dimensional with either concave or convex shape, which is pushed on a frictional surface by a group of robots. The object has M units of mass and I units of moments of inertial. By the assumption that there is no external vertical force on the object and a monotonous object contact with the surface, center of friction of the object is located on the center of gravity of the object (Fig. 2).

The amount of the frictional force between the object and the surface f_f is constant against the velocity of center of gravity (CG) of the object. The amount of the torque generated by the friction force is also constant with its direction against the direction of the angular velocity of the object.

Hence, kinematics equations of the object in existence of n number of different forces applied to the object is as follows:

$$M\vec{R} = \sum_{i=1}^n \vec{F}_i + f_f \frac{\vec{V}}{\|\vec{V}\|} \quad (1)$$

$$I\vec{\omega} = \sum_{i=1}^n \vec{r}_i \times \vec{F}_i + \tau_f \frac{\vec{\omega}}{\|\vec{\omega}\|} \quad (2)$$

where, \vec{V} is the speed vector of the object, $\vec{\omega}$ is the angular speed and τ_f is the torque that is applied to the rigid object.

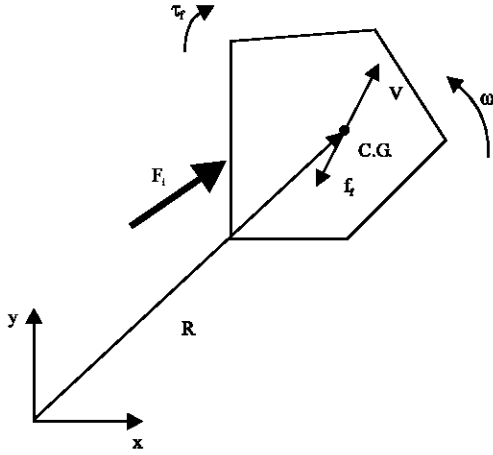


Fig. 2: A kinematic model of the object being pushed on a frictional surface

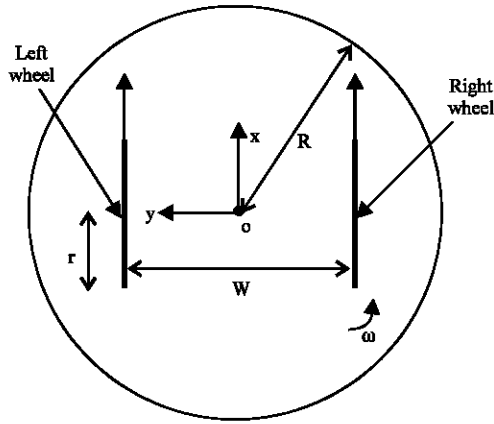


Fig. 3: A geometrical model of the robot and its corresponding parameters

Figure 3 shows a model of robots used in the simulations. Each robot has two parallel wheels with the same distance to the center of the robot. To eliminate the complexity of robot-object contact, it is assumed that the wheels of the robot reside inside the robot shape.

Therefore Eq. 3 holds, where, R is the radius of the robot, r is the radius of the robot wheels and w is the distance between the two wheels of the robot:

$$w^2/4 + r^2 < R^2 \quad (3)$$

The linear velocity of point O and the angular velocity of the robot are calculated as:

$$v_o = \frac{v_r + v_l}{2} \quad (4)$$

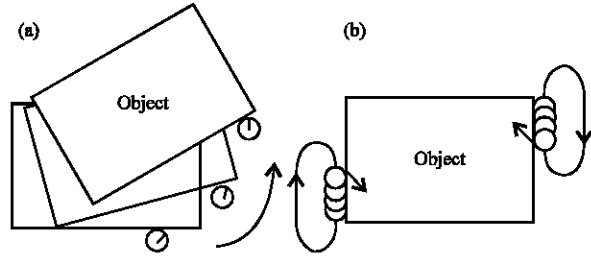


Fig. 4: The proposed method in Donald (1995) to the object pushing. (a) The rubbing behavior to reorient the object and (b) The rubbing behavior to reorient the object

$$\omega = \frac{v_r - v_l}{w} \quad (5)$$

PROPOSED METHOD

Considering the nonholonomicity of robots, in order to enable the robots to push the object while following object movements in all directions, it is necessary that the robots have variable contact point with the object. Hence, robots would rub against the object edge to apply a force while changing its relative position to the object as in (Fig. 4a).

Rubbing behavior is principally similar to what was introduced in Donald (1995), where they used this behavior to reorient the object. In Donald (1995) the robot compliantly applies a sliding force on the object edge until the end of the edge is reached, then the robot would turn around to reacquire contact with the object and repeats the same actions until the goal is reached (Fig. 4). To simplify the rubbing behavior when the robot contacts the object edge, it is always set to push the object with a constant relative angle to the object edge.

We can consider a gyrate queue of robots that do the same behavior with enough number of robots to cover the entire path that they move on (Fig. 5). This way, object movement would be speeded up due to the availability of at least one robot, at any time, in front of the object edge to apply a force. Assigning a particular queue of robots to each edge of the object enables the system to apply a force and torque to the object in any direction (Fig. 5).

To increase the robot utilization in the system it is better to merge the groups of robots in each queue into one group by making all the robots move around the object (Fig. 6). Unlike earlier explanations, each robot would move along the next edge if the end of current edge of the object is reached, instead of turning around and acquiring contact with the same edge of the object. Compared to the above paragraph, in this fashion all the

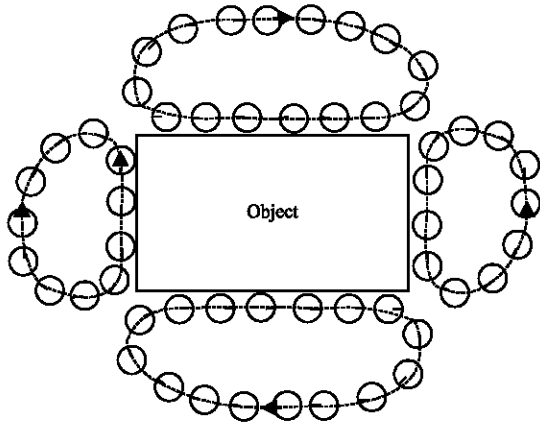


Fig. 5: Using groups of robots as an extension to the method in Donald (1995)

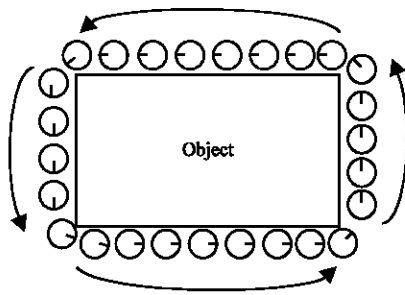


Fig. 6: Forming an orbit around the object in the proposed method

robots maintain their contact with the object providing more utilization in the system and taking less amount of robots. To easily devise a formation in the team of robots, it is decided the robots to rub or follow against the object edge only in one direction. Hence, the robots would form an orbit around the object rotating in counter clockwise direction (Fig. 6).

Based on the relative goal configuration to the object edge, robot decides to push the object using its rubbing behavior or to follow the object edge (Fig. 7). In case rubbing the object edge applies a force or a torque against the goal configuration, robot will follow the object movements instead. Following the object will help the robot to remain in the robots formation rotating around the moving object. To follow the object, a robot controller is designed to make it move along the object edge with a constant distance to the object edge. To make enough room for the possible object movements, robot will move along the object edge with at least a predefined distance to the object edge.

Decision making procedure of an individual robot is independent from other robots information. This attribute not only simplifies the controller but also makes the method independent from the number of robots available

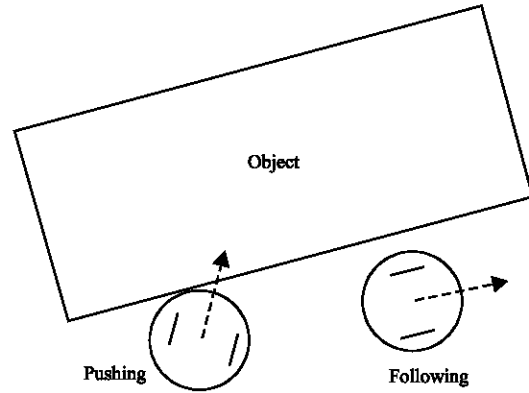


Fig. 7: A basic demonstration of the pushing and following behaviors of the robots

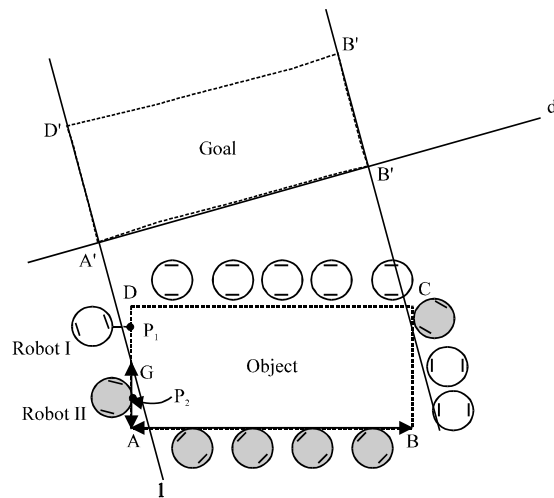


Fig. 8: A visualisation of the decision making algorithm and its corresponding parameters. The robots in gray are pushing while the robots in white are following the object

in the system. Each individual performs its action only based on the relative goal information and its relative position. In other words, it acts as if no other robot is available in the system.

Decision making procedure: Figure 8 shows the parameters which robots use to determine the proper action, whether to push or to follow. We define robot-object contact point as the closest point on the object edge to the robot, regardless of having a physical contact between the robot and the edge. In Fig. 8 contact points for Robot I and Robot II are shown by P_1 and P_2 , respectively.

To determine whether pushing the object may result in a proper object movement toward the goal, the relative position of contact point to goal is used. Robot will push

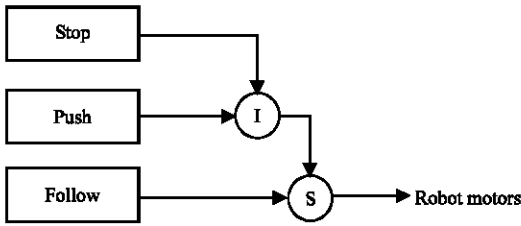


Fig. 9: The basic behavioral layers for manipulation

the object only if its contact point is placed between the robot and the extended corresponding edge of the goal. For instance, as P_2 is between the Robot II and the line l , Robot II decides to push the object while Robot I will only follow the object as P_1 is placed in the opposite side of line l . Figure 9 shows how the two basic behaviors relate to each other in subsumption architecture.

In the instance shown in Fig. 8, robots that move along AD will follow the object edge while their contact position is placed on the line DG . They push the object only when their contact position is on AG . Accordingly, robots on the edge AB are pushing the object. Robots painted in gray in Fig. 8 have selected their Pushing behavior while others are following the object.

Using this scheme, the groups of robots on any edge of the object tend to align the edge to the corresponding edge on the goal. It is obvious if all the edges of the object get aligned with their corresponding edges on the goal, the object will be placed on the desired goal position. Therefore, as the decision mechanism does not have any consideration to the object friction center and center of gravity of the object, it makes the algorithm independent from the object model and its shape.

Controller design: As robots perform in a continuous state and action space, fuzzy controllers are chosen to determine robots actions. Each robot performs individually in the system. In other words, the information about other robots is not included in the state space of each individual robot. Two separate fuzzy controllers are designed for the two main robot behaviors: namely, pushing and following.

As we considered this object manipulation system is a quasi-static one, the linear speed and the angular speed of the object are not included in the state space of the fuzzy system.

The state space for the pushing and following actions are relative robot angle to the object edge α , robot distance to the edge (d) and relative robot position to the left point of the object edge (l) as shown in equation 6. Figure 10 shows a demonstration of these parameters.

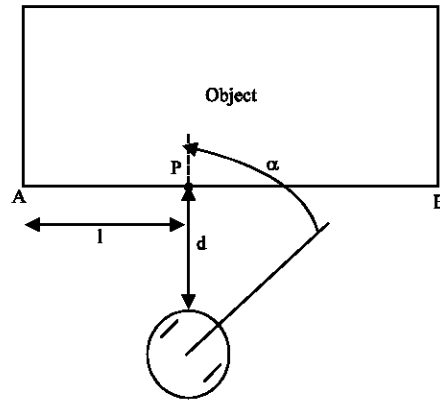


Fig. 10: A demonstration of parameters used as inputs for the fuzzy controllers

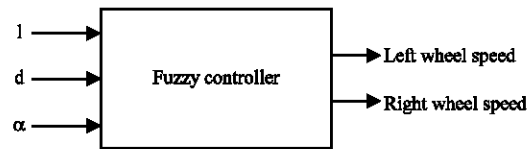


Fig. 11: A schematic of fuzzy controllers used for pushing and following behaviors

To make the fuzzy controller independent from the length of the object, the value of parameter l is mainly calculated proportional to the edge length. Hence, the value of l is calculated as in equation 7. Consequently, the inputs to the fuzzy controllers are a , d and l as shown in Fig. 11.

$$S = (l, d, \alpha) \tag{6}$$

$$l = \frac{|AP|}{|AB|} \tag{7}$$

Additionally, actions space is selected to be the robot's left and right wheel velocities (V_l and V_r) as shown in Fig. 11.

As robots directly get the desired left and right wheel velocities from the fuzzy controllers, the need for a path planner is eliminated in the system. The nine possible pairs as the consequent of fuzzy rules in the fuzzy controllers are shown in Fig. 12. Each cell in the figure represents the possible consequent of a rule in the fuzzy controller.

The fuzzy sets used in the fuzzy controller are shown in Fig. 13. According to Fig.13a, the input range for dimension l is partitioned into three nonoverlapping interval. The $(-\infty, 0)$ interval helps in defining rules to make the robot move toward the edge when it is not in front of

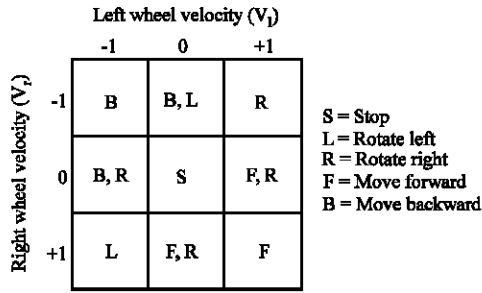


Fig. 12: The action space of the system. The nine pairs of values shown are used as fuzzy rule consequents

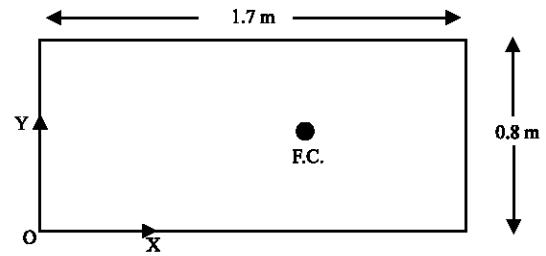


Fig. 14: The geometrical model of the object used in the simulations

SYSTEM SIMULATION

Simulation model: There are some assumptions to simulate the system. The object model used in the simulations is a two dimensional object of 1.7×0.8 m size and 5 units of mass and 0.5 units of moment of inertia (Fig. 14). The object coordinate frame is placed at O. The object reference point is also located at point O. We utilized 20 robots in the simulations with $R = 15$ cm radius, in which the distance between the wheels are $w = 23$ cm. The static and dynamic friction coefficients between the object and ground are $\mu_s = 0.8$ and $\mu_t = 0.3$, respectively.

To model the force interaction between the object and each robot we have used spring and damper mechanism. The force is decomposed to a frictional force and a normal force. Damping factor and spring coefficient used as $B = 1$ and $K = 1k$ to simulate frictional force and $B = 100$ and $K = 20 k$ for normal force, respectively.

Simulating object and robots force interaction: To simulate the elasticity of the object, edge deforming in the contact point and the interaction force can be simulated by a spring and a damper. As the robot pushes the object or in other words moves inside the object edge in a microscopic scale, it is considered that the object-robot contact is a spring-damper system.

Considering the robot has a contact with the object at point A (Fig. 15), it will move to the point B when it moves inside the object. In other words, vector \overline{AB} is the displacement of the contact point. This displacement vector can be decomposed to two vectors; \overline{Y}_x and \overline{Y}_y which are the projections on \overline{X} and \overline{Y} , respectively. It is shown in Eq. 8 and 9.

$$\overline{Y}_x = \overline{AB} \cdot \overline{X} \tag{8}$$

$$\overline{Y}_y = \overline{AB} \cdot \overline{Y} \tag{9}$$

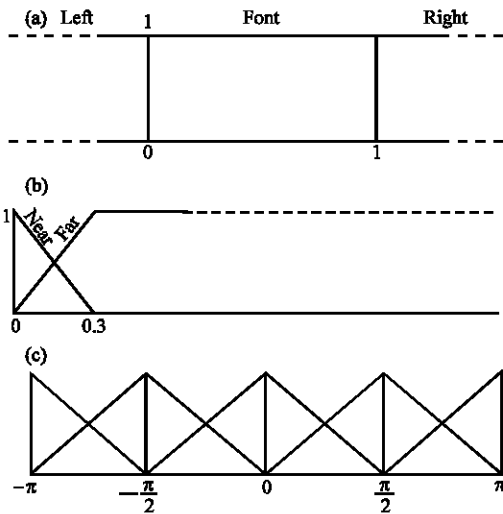


Fig. 13: The fuzzy sets used in the fuzzy controllers for the pushing and following behaviors. (a) Edge position, (b) Edge distance and (c) Relative angle

the corresponding edge. Similarly, the $(0, \infty)$ interval enables to define rules for the situations when the robot has passed the object edge. Moreover, when the value of l is between $(0, 1)$, the robot would be in front of the object edge. Finally, having a fuzzy set that covers the infinity for the input range d (Fig. 13b) enables the robots to approach the object when they are far from it. The sets used to partition a are shown in (Fig. 13c).

When a robot decides to push or to follow, its controller will control the chosen behavior. For following behavior, the controller makes the robot to move along the edge that robot is assigned to. It does not differ that the robot is in far or near interval. But when the robot decides to push, its controller makes it to get closer to the edge of the object. When the robot is in the interval of near, the controller tries to maintain angle a in a constant amount ($\pi/5$ in this study).

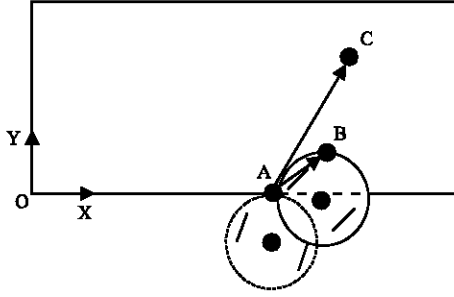


Fig. 15: Movement of the contact point makes the vector \overline{AB}

The speed vectors, the derivation of vectors \overline{Y}_x and \overline{Y}_y are \overline{V}_x and \overline{V}_y , respectively, (Eq. 10 and 11).

$$\overline{V}_x = \frac{d\overline{Y}_x}{dt} \quad (10)$$

$$\overline{V}_y = \frac{d\overline{Y}_y}{dt} \quad (11)$$

At last frictional force and normal force are \overline{F}_x and \overline{F}_y , respectively; (Eq. 12 and 13).

$$\overline{F}_x = \overline{Y}_x \cdot K_x + \overline{V}_x \cdot C_x \quad (12)$$

$$\overline{F}_y = \overline{Y}_y \cdot K_y + \overline{V}_y \cdot C_y \quad (13)$$

where, C_y and K_y are damping factor and spring coefficient of \overline{F}_y , while damping factor and spring coefficient of \overline{F}_x are C_x and K_x , respectively. If Eq. 14 holds,

$$|\overline{F}_x| > |\overline{F}_y| \cdot \mu_s \quad (14)$$

Friction force (F_f) will be as Eq. 15.

$$|\overline{F}_f| = |\overline{F}_y| \cdot \mu_d \quad (15)$$

where, μ_s is static friction coefficient and μ_d is dynamic friction coefficient between each robot and the object. The direction of vector \overline{F}_f is opposite the direction of vector \overline{F}_x .

Finally the applied force and torque that effect on the object are \overline{F} and $\overline{\tau}$, respectively (Eq. 16 and 17).

$$\overline{F} = \overline{F}_x + \overline{F}_y \quad (16)$$

$$\overline{\tau} = \overline{F} \times \overline{AC} \quad (17)$$

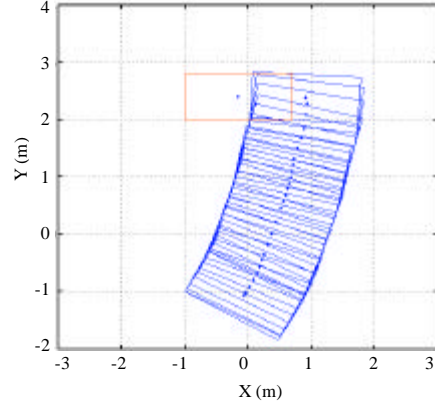


Fig. 16: Only a part of the manipulation action, which is to align just the bottom edge of the object to the corresponding edge of the goal, has been done

Simulation results

Simulation 1: Aligning a single object edge to the goal:

To proof that system can align a single arbitrary edge of the object to the corresponding edge on the goal, a number of simulations were performed. In this simulation we temporarily modified the algorithm to only follow the object edges except one edge. In other words, the robots just try to align a particular edge of the object to the corresponding edge on the goal.

To achieve this behavior, the robots decide to push or follow the edge based on relative goal position explained earlier. In simulation result shown in Fig. 16 robots are instructed to follow the object on the left, right and top edge of the object. It can be seen that robots could align a single edge, the bottom edge in this example, of the object to the goal. Due to the undesirable errors in the system, position error and orientation error are not completely compensated. This is because robots do not push the object on the other edges. Hence, this disables the system to negate the extra object movements if the object passes by the goal edge.

Simulation 2: Pushing the object toward the goal:

In the simulation shown in Fig. 17 all edges of the object are set to be pushed if necessary according to the proposed method. The figure shows the result of commanding the group of robots to move the object from point (1, -1) to (-1, 2). The friction center is located on point (0.85, 0.4) in object coordinate frame.

It is obvious in Fig. 17b, c that the position error and orientation error are monotonically decreasing to an acceptable value.

If the difference of early orientation of the object to the goal orientation does not exceed of being little, the system can move the object to any desired goal.

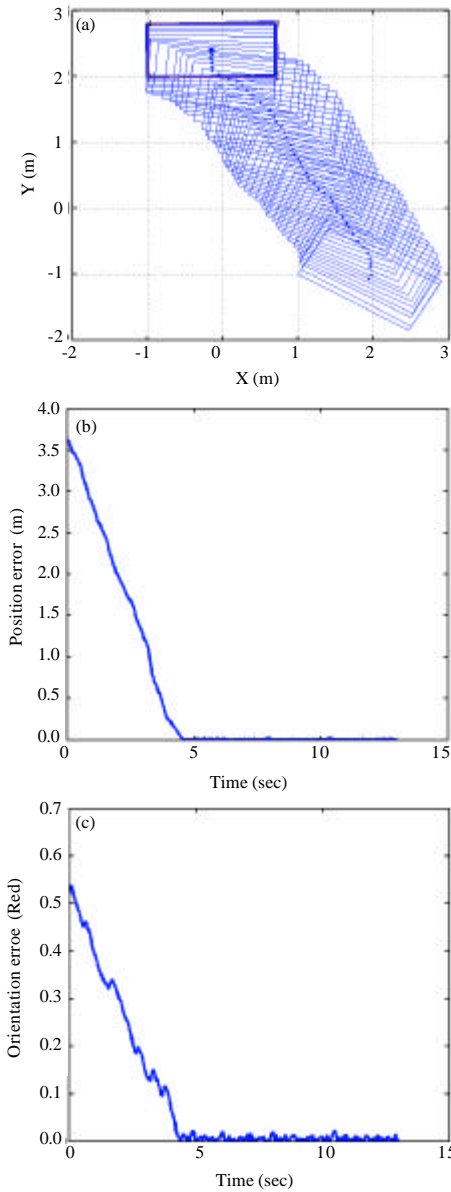


Fig. 17: Track of object being pushed toward the goal using the proposed method. (a) The object friction center is located on the geometrical center of the object. (b) Position error and (c) Orientation error

In this simulation and the others, the goal is not near to the object but their orientations are near in value.

Simulation 3: Pushing an object with an arbitrary center of friction: To test the independency of the method to the location of friction center, the simulation result presented in simulation 2 is accomplished with a modification to the

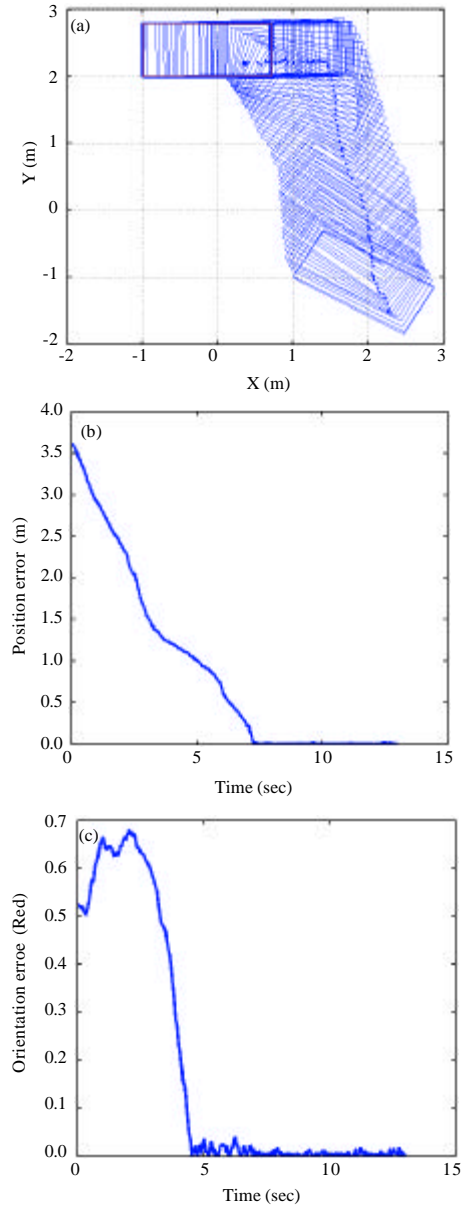


Fig. 18: Track of object being pushed toward the goal using the proposed method. (a) The object friction center is located on an arbitrary location other than the geometrical center of the object. (b) Position error and (c) Orientation error

position of the object friction center. In this simulation, the friction center is placed at (1.3, 0.2) in object coordinate frame (Fig. 18).

Comparing results with earlier simulation, increasing in the orientation error occurs at first (Fig. 18c). This is because of placement of the friction center close

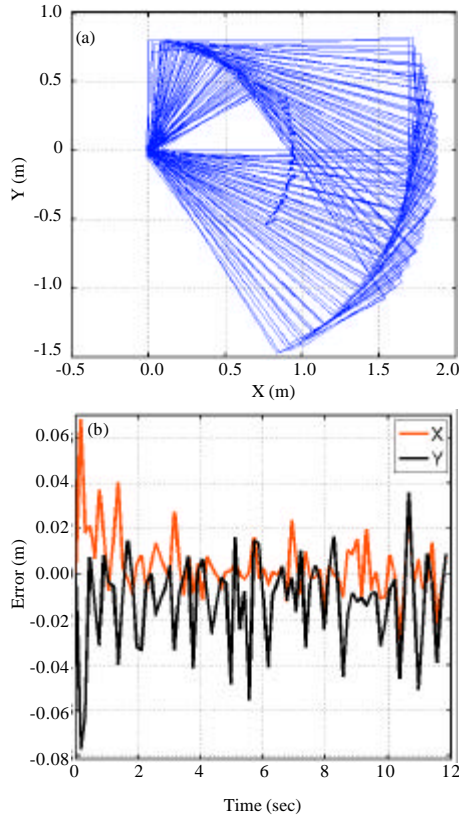


Fig. 19: Track of the object being rotated around an arbitrary point on the object. The group of robot were able to rotate the object around the center of rotation with a small displacement. (a) Object track and (b) Locus of the center of rotation

proved. As a result of this ability and potency to transfer to the corner of the object. Hence, any force applied to the object may cause a great value of torque on the object which may lead to undesirable object rotations. But it can be seen that the system can compensate this undesirable object movements, leading the object toward the goal providing sufficient accuracy. As it is shown in (Fig. 18b), the position error has decreased to an acceptable value.

Simulation 4: Rotating the object around an arbitrary point: To test the drivability of the system, it is decided to rotate the object around an arbitrary point on it. Figure 19a shows the track of the object being rotated around the point (0, 0) in the object coordinate frame.

Figure 19b shows the displacement of the desired center of rotation; point (0, 0). It can be seen that robots could rotate the object around the center of rotation with quite small displacements.

By this simulation, ability of the system to turn the object through any desired orientation trajectory is

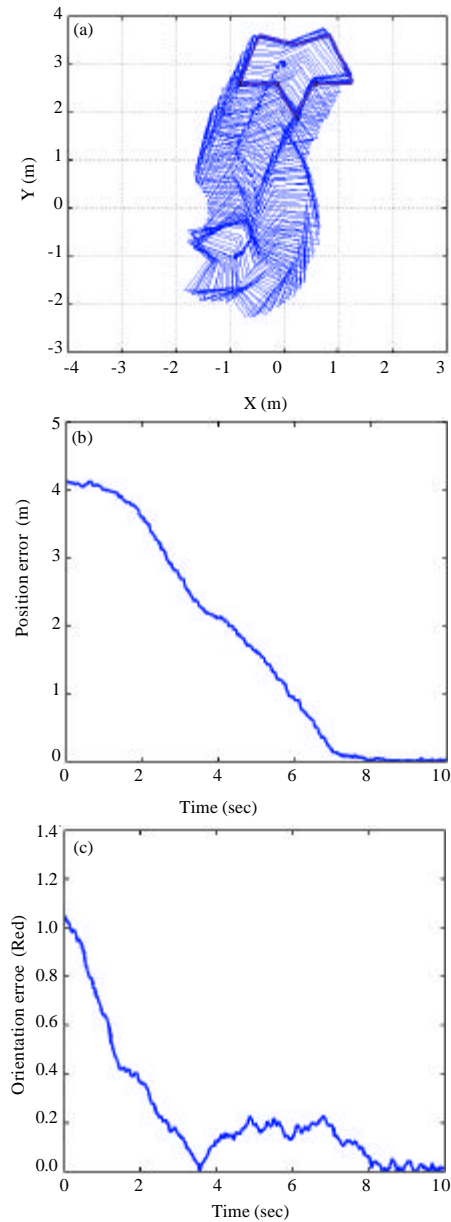


Fig. 20: Result of applying the proposed method to an object of concave shape. (a) The track of the object is shown while being pushed from (-1, -1) to (0, 3). (b) Position error and (c) Orientation error

the object toward a goal, as it was shown before, this system can move the object through any desired trajectory.

Simulation 5: Pushing an object of an arbitrary shape: A couple of simulation was performed to validate whether

the proposed method can be applied to any object of an arbitrary shape. Figure 20 shows an example with an object of concave shape and its track while being pushed from (-1,-1) to (0, 3). As it can be seen in the Fig. 20, the object is successfully pushed to the goal location. The results show the independency of the method to the object shape without any modifications to the controller and parameters in the system. In this simulation the number of robots is set to 30. It is obvious in Fig. 20b that the position error is monotonically decreasing to an acceptable value. As it is shown in Fig. 20c the orientation error is decreased to a small value at the end of simulation.

CONCLUSION

A new swarm approach to transport a polygonal object is introduced. Based on the proposed idea, robots form a rotating orbit around the object to surround and move the object as the orbit moves toward the goal. In this method, each robot performs individually in the system without a need to the information from the other robots in the system. The robots communicate using stigmergy communication.

Based on this idea, using a simple decision making, each robot could easily determine its proper action to push the object while remaining in the robot formation. Fuzzy controllers were used to develop each behavior in each robot.

Two major behaviors of the robots are described. They have been designed in such a way that enables the robot to remain in the robot formation during object manipulation.

The main characteristics of the developed system were independency of the method to the object kinematics and its shape. Moreover, the method was not dependent on the number of robots available in the system. The system has the ability of rotating the object in an acceptable result. Moreover, it can transfer the object to any desired goal, by a limited difference of orientation. Simulation results were given to support the proposed approach. So the system can move the object through any desired trajectory.

Future research to this system include providing experimental results of the system performance. Additionally, we would like to mathematically analyze and proof the method. Modifying the *Pushing* behavior to push the object with variable relative angle and compare the performance is another study in the future.

REFERENCES

Ahmadabadi, M.N. and E. Nakano, 2001. A constrain and move approach to distributed object manipulation. *IEEE Trans. Rob.*, 17: 157-172.

Aminaeie, A.H. and M.N. Ahmadabadi, 2007. Learning cooperative object pushing with variable contact point. *Proceedings of the IEEE/ASME International Conference on advanced Intelligent Mechatronics (AIM)*, September 4-7, Zurich, pp: 1-6.

Bernheisel, J. and K.M. Lynch, 2005. Stable pushing of assemblies. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, April 18-22, Barcelona, Spain, ISBN: 0-7803-8914-X, pp: 3280-3287.

Bishop, B.E., 2008. Swarm-based object manipulation using redundant manipulator analogs. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2008, Pasadena, CA., pp: 1495-1500.

Bjerknes, J.D., A.F.T. Winfield and C. Melhuish, 2007. An analysis of emergent taxis in a wireless connected swarm of mobile robots. *Proceedings of the IEEE Swarm Intelligence Symposium (SIS)*, April 1-5, IEEE, pp: 45-52.

Bonabeau, E., M. Dorigo and G. Theraulaz, 1999. *Swarm Intelligence: From Natural to Artificial Systems*. 1st Edn., Oxford University Press, New York, ISBN: 0-19-513159-2, pp: 183-205.

Chaimowicz, L., V. Kumar and M.F.M. Campos, 2004. A paradigm for dynamic coordination of multiple robots. *Auton. Robot. J.*, 17: 7-21.

Donald, B., 1995. On information invariants in robotics. *Artif. Intell. J.*, 72: 217-304.

Donald, B., L. Gariepy and D. Rus, 2000. Distributed manipulation of multiple objects using ropes. *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, April 2000, San Francisco, CA, USA., pp: 450-457.

Fink, J., N. Michael and V. Kumar, 2007. Composition of Vector Fields for Multi-Robot via Manipulation Caging. In: *Robotics: Science and Systems*, Burgard, W., O. Brock and C. Stachniss (Eds.). The MIT Press, Atlanta.

Groß, R., F. Mondada and M. Dorigo, 2006. Transport of an object by six pre-attached robots interacting via physical links. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, May 15-19, IEEE Organization, pp: 1317-1323.

Haverinen, J., M. Parpala and J. Roning, 2005. A miniature mobile robot with a color stereo camera system for swarm robotics research. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Apr. 18-22, Barcelona, Spain, pp: 2494-2497.

- Hirata, Y., K. Kosuge, H. Asama, H. Kaetsu and K. Kawabata, 2000. Coordinated transportation of a single object by multiple mobilerobots without position information of each robot. Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), Nov. 2000, Takamatsu, Japan, pp: 2024-2029.
- Maneewarn, T. and P. Detudom, 2005. Mechanics of cooperative nonprehensile pulling by multiple robots. Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), Aug. 2005, IEEE, pp: 2004-2009.
- Martinoli, A., K. Easton and W. Agassounon, 2004. Modeling swarm robotic systems: A case study in collaborative distributed manipulation. *Int. J. Rob. Res.*, 23: 415-436.
- Michael, N., J. Fink and V. Kumar, 2008. Experimental testbed for large multirobot teams. *IEEE Rob. Autom Mag.*, 15: 53-61.
- Mondada, F., L.M. Gambardella, D. Floreano, S. Nolfi, J.L. Deneuborg and M. Dorigo, 2005. The cooperation of swarm-bots: Physical interactions in collective robotics. *IEEE Rob. Autom Mag.*, 12: 21-28.
- Pettinaro, G.C., L.M. Gambardella and A. Ramirez-Serrano, 2005. Adaptive distributed fetching and retrieval of goods by a swarm-bot. Proceedings of the International Conference on Advanced Robotics (ICAR), July 2005, Seattle, WA, pp: 825-832.
- Sahin, E., 2004. Swarm robotics: From sources of inspiration to domains of application. *Lecture Notes Comput. Sci.*, 3342: 10-20.
- Sharkey, A.J.C., 2006. Robots, insects and swarm intelligence. *Artif. Intell. Rev.*, 26: 255-268.
- Steele, F. and T. Geb, 2007. Directed stigmergy-based control for multi-robot systems. Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction, March, Arlington, Virginia, USA., pp: 223-230.
- Stilwell, D.J. and J.S. Bay, 1993. Toward the development of a material transport system using swarms of ant-like robots. proceedings of the IEEE International Conference on Robotics and Automation (ICRA), May, Atlanta, GA, USA., pp: 766-771.
- Wang, Z., E. Nakano and T. Matsukawa, 1994. Cooperating multiple behavior-based robots for object manipulation. Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS), September, Munich, Germany, pp: 1524-1531.
- Wang, Z., Y. Hirata and K. Kosuge, 2005. Deformable caging formation control for cooperative object transportation by multiple mobile robots. Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), July, Monterey, CA., pp: 1158-1163.
- Yang, X., K. Watanabe, K. Kiguchi and K. Izumi, 2003. Coordinated transportation of a single object by two nonholonomic mobile robots. *Artif. Life Rob.*, 7: 48-54.
- Zhang, D., G. Xie, J. Yu and L. Wang, 2007. Adaptive task assignment for multiple mobile robots via swarm intelligence approach. *Robot. Auton. Syst.*, 55: 572-588.