A New Approach to Manipulate Objects with a Team of Distributed Robots Based on Constrain-and-Move Strategy

Mohammadreza Barouni-Ebrahimi
Computer Science Department
University of New Brunswick
Fredericton, Canada
m.barouni@unb.ca

Nasser Ghariban
Engineering and Technology Department
Virginia State University
Petersburg
nghariba@vsu.edu

Abstract

Based on constrain and move concept, a new algorithm CCM (complex Constrain and Move) is proposed to reorient objects with a distributed object handling mobile robots. In traditional constrain and move strategy, a group of robots constrain the object in undesirable directions and another group push the object to move it in desired path. In proposed CCM method, each robot does both constrain task and move task and the algorithm of robots are the same. A series of dynamic computer simulations are conducted to measure the efficiency of the new proposed algorithm for different experiments. The results of the simulations indicate that the system is stable in all experiments and it is more faults tolerant. Simulations also revealed that the minimum necessary numbers of robots in this method to handle the object is less than traditional constrain and move strategy.

Introduction.

A strong, complicated and expensive robot is needed to manipulate a heavy, big object. Such a robot usually depends on the shape of the object. The cooperation of some simple and cheap robots is a suitable way to manipulate objects with various shapes. Different controlling methods exist in this area of researches, Such as: a) Leader-follower [1] b) Centralized [2] c) Distributed [3], which are the three main controlling methods for cooperative object manipulating systems.

Deep dependence of robot teams to leader (Leader-follower) or central unit (Centralized systems) causes to decrease fault tolerance. Although performance in distributed systems is not dependent to one special robot, making coordination in such systems has its own complications.

Constrain and Move strategy becomes mooted to rotate an object around a desired point by a distributed robot teams in [2]. In this method, a group of robots constrain object in undesirable directions and another group push the object. The result of this cooperation is the rotation around the desired point.

The divisions of robots into two separated teams, which are, constrain and move is the weakness of implementation of this method. Also the existence of some errors in position of robots causes the object to be locked.

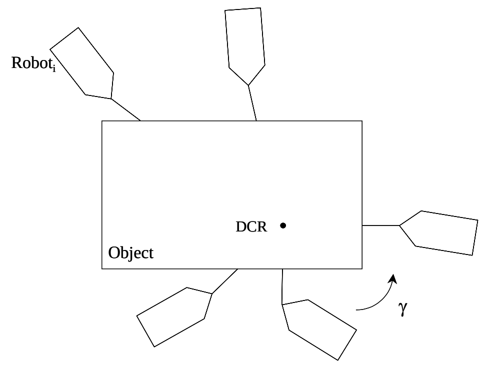

Figure 1: Transferring an object with multiple robots

Detection and solving the faults in [4] for position errors of the robots have bean considered. In this paper, a distributed method is proposed for rotating an object with a team of cooperative object handling mobile robots. In the algorithm each robot does both constrain and Move tasks and the algorithm of robots are the same.

In the next section the definition of problem and assumptions are introduced, then Constrain and Move strategy for rotating an object is explained. The Complex Constrain and Move (CCM) algorithm is proposed in fourth section. Simulation results are defined in section 5. Final remark is explained in the last section.

Problem Definition

In this paper we consider rotation as a part of transferring object. Assume we wish to turn a two dimensional object (figure1) around point DCR. Some robots that know the desired center of rotation (DCR) constrain the object's rotation center while the other robots rotate it. Each robot will run the same rotation algorithm. If the minimum necessary numbers of robots are available (3 robots) and the angle between each neighboring pairs of robot arms is less than 180 degrees then the cooperation of distributed object handling mobile robots will be caused to rotate the object around DCR point with consideration of some ignorable errors.

Assumptions

Each object handling robot has an arm with one degree of free. The robot arms are modeled with springs and dampers (figure2).

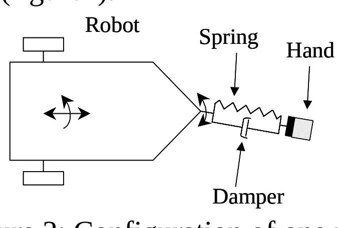

Figure 2: Configuration of one robot of the team

The force applied by each robot is the sum of the force applied by spring (F_{spring}) and damper (F_{damper}).

$$\vec{F}_i = \vec{F}_{spring} + \vec{F}_{damping}$$

$$\vec{F}_{spring} = K \times \Delta l \ , \ \vec{F}_{damping} = B \times \Delta V, \ \Delta V = \frac{\Delta l}{\Delta t} \quad (1)$$

Where K is the spring stiffness and Δl is spring deflection. B is the damping factor and ΔV is the difference of the velocities of the head and the end of the spring.

Each robot controls the compression of its arm spring and the arm-object angle with its sensors.

See figure3, the parameters are defined as below:

1- DCR : Desired Center of Rotation point
2- L_{0i} : Free length of the spring of robot$_i$ arm
3- L_i : Length of the spring of robot$_i$ arm
4- α_{oi} : Arm-object angle of robot$_i$ when robot arm direction passes through DCR
5- α_i : Arm-object angle of robot$_i$
6- H_i : Head of the robot$_i$
7- d_{0i} : Distance between DCR and H_i when $L_i = L_{0i}$
8- d_i : Distance between DCR and H_i
9- C_i : Arc of the robot$_i$ when DCR is the center and d_{0i} is the radius
10- γ : Rotate direction, -1 is clock wise direction and +1 is its opposite
11- θ_{0obj} : Initial object angle
12- θ_{obj} : Object angle
13- θ_{des} : Desired object angle

The *Constrain-Move* strategy for reorientation of an object in place

When a train moves on rails, the rails interaction forces constrain the train so that it can only move along them. When a crank turns, the reactive wrenches of the ground cancel out all of the external forces about the crank-ground joint.

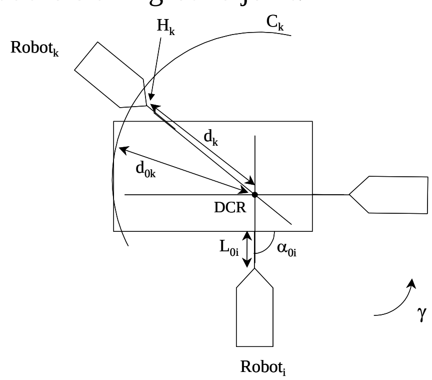

Figure 3: Parameters of the system the resulting torque

In these examples, the robot or the operator is not concerned about constraining the object. Therefore, the main job is to control the object along its desired path in such a way that the interaction forces do not exceed their limits or a jamming does not occur. This idea is used to develop the Constrain-Move strategy [11]. In this strategy a group of the robots constrain undesirable movements of the object and another team of the robots pushes the load on desirable path.
Figure4 shows object rotation around point O [10]. Three robots (white ones) constrain the object's rotation center. The desired rotation center (O) does not move and the object is free to rotate. The black robot pushes the object with force F. F_y creates a torque around point O and the object rotates.

This strategy is done for moving the object on a straight line [3] and moving on desired path [5]. If there are some errors in constraining robot arms directions, the lock problem will be occurred [10,11]. Compliance of the robot arms is needed for partially solving lock problem. Weakness of compliance leads to propose methods for detection and solve lock problem. We have two separated teams in all proposed methods; Constrain team and Move team. On the other hand, we need at least 4 robots for rotation. Also tasks of the agents are not the same. This new method is considered for removing mentioned problems.

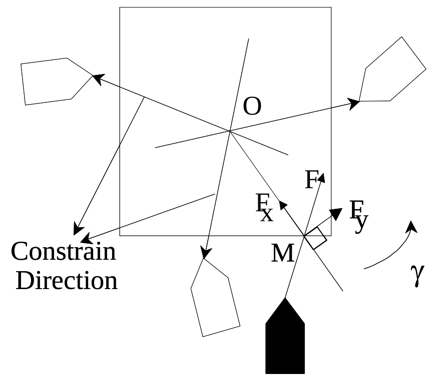

Figure 4: Turning an object with a team of robots (C&M algorithm)

CCM (Complex Constrain-Move) Method

This method is based on Arm Rotation Method [10]. Each agent of the team does both constrain and move tasks; therefore all robots have the same task. It supports system to increase fault tolerance by using redundant robots. Rotation of an object can be done with the minimum number of robots (3 Robots).

Lets consider one of the robots, see figure5. Since $d_i \le d_{0i}$,the locus of possible rotation center of the object is in the left half-plane of each robot arm, when the object is rotated in γ direction. Therefore, the object instantaneous center of rotation (ICR) must be inside the area where the entire one-robot-object rotation center locus overlaps. We call this area "the Possible Rotation Center Area (PRCA)".

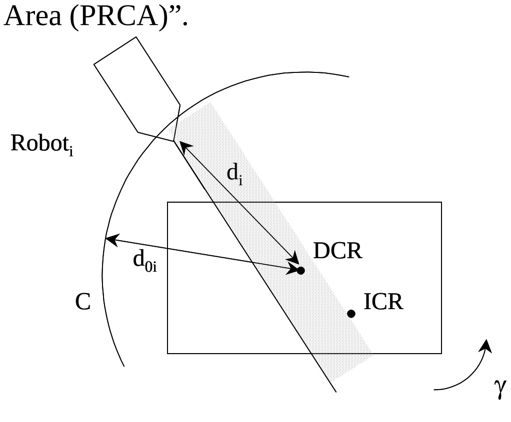

Figure 5: PRCA for each robot

Due to the γ direction in figure6, when robot 1 is in state 2, PRCAs of the robots overlap at $M_1M_2M_3$ (common triangle). ICR will be in this triangle. In state 1 there isn't any common triangle, so the object can't turn in γ direction and the lock problem occurs.

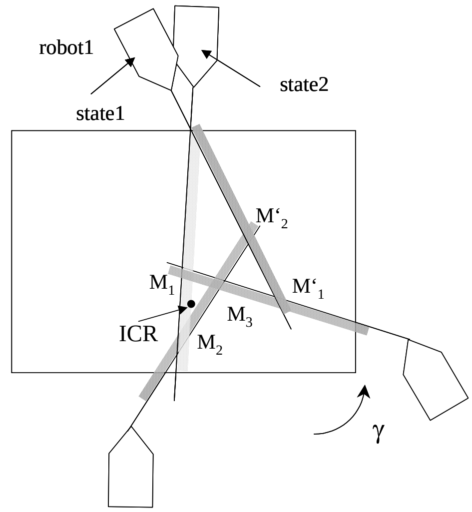

Figure 6: Common triangle $M_1M_2M_3$ will create when robot1 is in state2

This problem is discussed in [10]. New algorithm is proposed according to common triangle idea.

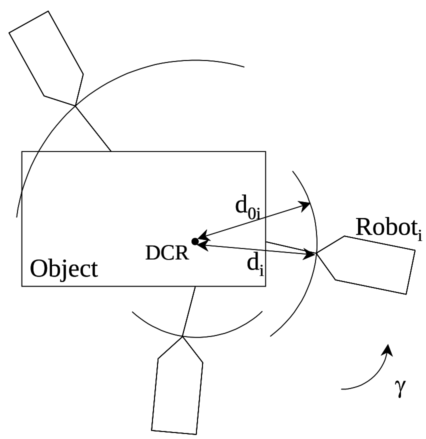

Figure 7: Team of the robots do the constrain task

Figure7 shows object constraining task. Robots move in a way that in all situations $d_i <= d_{0i}$ so the algorithm does the constrain task. Figure 8 shows the object Move task. Each robot pushed a controlled force to object. F_{yi} creates a torque around point ICR and the object rotates. Figure 9 shows the combination of these two tasks. Each robot controls its arm-object angle and amount of the force that pushes the object. Due to DCR position and γ direction, each robot controls its arm angle to create common triangle $M_1M_2M_3$ with overlapping PRCAs. DCR point is in this triangle. With F_1 ,F_2 and F_3 forces, ICR point will always be in $M_1M_2M_3$ triangle. Sum of the robot forces for each ICR point among mentioned triangle makes necessary torque. On the other hand, the common triangle is small, so ICR point is close enough to DCR point and the error can be ignored (This error is mentioned in simulation result). Based on the algorithm, each robot ($robot_i$) controls its arm-object angle (α_i) and arm spring length (L_i) as below:

$$L_i = L_{0i} - C_1|\theta_{dif}|$$
$$\alpha_i = \alpha_{0i} - C_2\theta_{dif} \qquad (2)$$

where, C_1 and C_2 are constant, $\theta_{dif} = \theta_{des} - \theta_{obj}$.

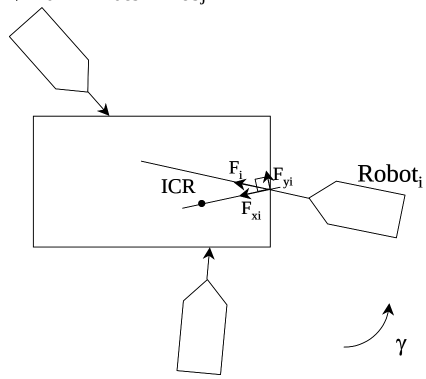

Figure 8: Team of the robots do the move task

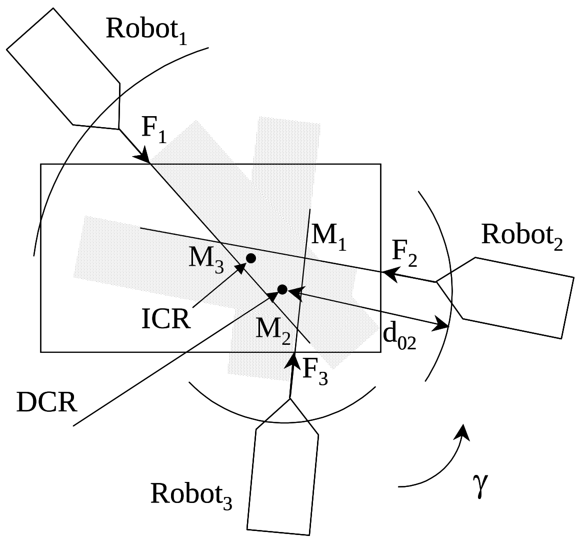

Figure 9: Turning an object with CCM algorithm

Constrain task is done by each robot with this condition:

If $d_i > d_{0i}$ then
  $L_i = L_{0i} – (d_i – d_{0i})$ (3)

The brake system is simulated with changes in the amount of C_1 and C_2. Some errors in arm angle of the robot (α_0) can be handled automatically. Simulation results show that the algorithm is efficient.

Two measures

Two parameters are considered to check the efficiency of the algorithm. The first one is displacement of the desired rotation center in the world coordinate system and is defined as:

$$e_r = \sqrt{\Delta x_{d.c.r.}^2 + \Delta y_{d.c.r.}^2} \qquad (4)$$

where $\Delta x_{d.c.r.}$ and $\Delta y_{d.c.r.}$ are displacements of the desired center of rotation (DCR) in X and Y directions of the world coordinate system.

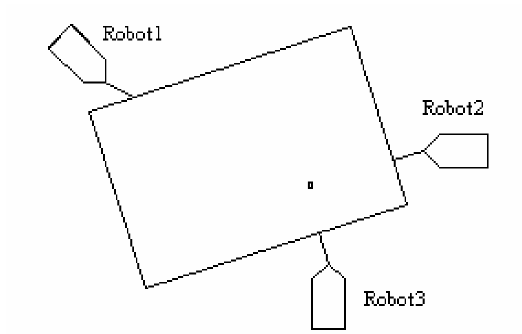

Figure 10: Geometrical model of the simulated system with 3 robots

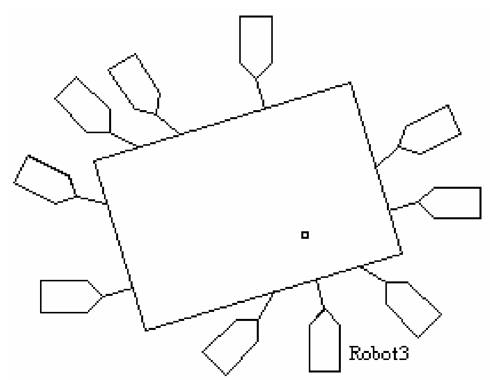

Figure 11: Geometrical model of the simulated system with 10 robots

The second parameter is the interaction forces between robots and object. These forces are a function of the arm stiffness.
We define F_r as average of the interaction forces:

$$F_r = \frac{\sum_{i=1}^{n} k_i \Delta x_i}{n} \qquad (5)$$

where K_i is the stiffness coefficient of the ith robot arm, Δx_i is its deflection and n is the number of robots.

Simulation results

A series of dynamic computer simulations is conducted to study effects of the CCM algorithm.
Figure10 shows the model used in the simulations. In this model, the object is a (0.6,0.4) square of unit mass and moment of inertia. The mass center is placed at the geometrical center of the object. The desired center of rotation is at (0.1, 0.1) of the mass center of the object. The robots (robots 1, 2, and 3) are located at (-0.3, 0.3), (0.2,0.0), and (0.0, -0.1) of the desired center of rotation respectively. The stiffness and damping factors are set to 2000 N/m and 200 Ns/m. The maximum force applied by the robots is set to 40N. The initial orientation of the object is set to 0 degrees and the goal is set to 30 degrees. Figure11 shows the model with ten robots. The object movement for a sample test shows in figure 12.

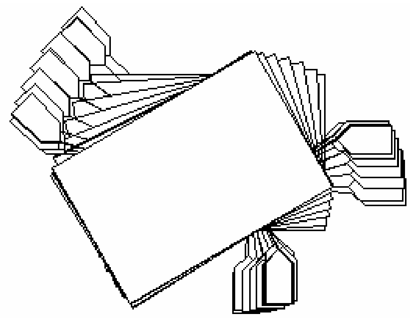

Figure 12: Object movement in turning

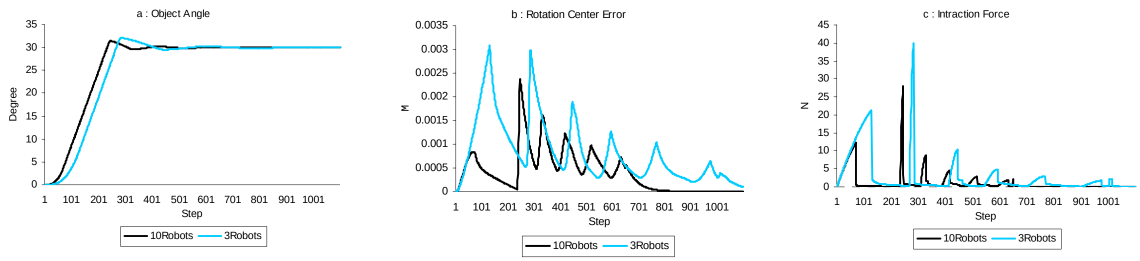

Figure 13: Sim.1, Robots without any error

We describe two simulations in figure13, figure14. In sim.1 there aren't any errors in robot arm angles (α_{0i}), but In sim.2 there is 6 degrees error in robot3 arm angle. In each simulation we have done two experiments, one of them is done bye 3 robots and the other one is done bye 10 robots. It can be observed that the object has been rotated to the desired angle (30 degrees) and the system is stable in all experiments. We see that the maximum e_r is ignorable in comparison with object size. Maximum value of F_r is arisen because of the braking system. Comparing two simulations, we can see that in sim.2 the average value of e_r and F_r for 3 robots experiments is greater than those in sim.1 but for 10 robots experiments those values are nearly the same. That means increasing

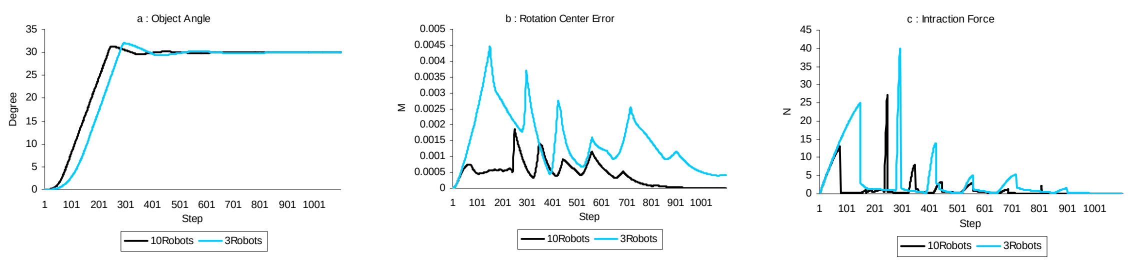

Figure 14: Sim.2, Robot3 has 6 degrees error in arm angle

the number of robots can cover the error of a robot arm angle

With comparing experiments of 3 robots and 10 robots, we have greater values for e_r and F_r in 3 robots.

In sim.1 the maximum value of e_r for 3 robots reaches to 0.003M. In both experiments the final value of e_r reaches to zero. The maximum value of F_r is 40N.

In sim.2 the maximum value of e_r is 0.005M and the final value doesn't reach to zero in simulation steps for 3 robots.

Final remarks

In this paper Complex Constrain and Move (CCM) method is introduced to rotate an object with a team of distributed object handling mobile robots. This method is based on the Constrain and Move strategy. In this method all of the robots have the same algorithm. Each robot controls its force to the object and its arm-object angle to do the algorithm. To constrain an object, we need at least 3 robots and these number of robots are sufficient to move the robot too. This algorithm handles some errors in the position of robots automatically.
The simulation results show that increasing of some robots in system give better result and increase the fault tolerance. Transferring an object in a straight line or arbitrary path with this algorithm must be studied.

References

[1] Nakano E. et. al, " Cooperational control of the anthropomorphous manipulator MELARM, " *Proc. of 4th Int. Symp. on Industrial Robots,* pp. 251-260,1974

[2] Y. Nakamura "Advanced robotics: Redundancy and optimization", *Addison-Wesley, 1990*

[3] Majid Nili Ahmadabadi and Nakano Eiji. "Task allocation and distributed cooperation strategies in a group of object transferring robots" *In Proc. of the 1997 IEEE/RSJ Int. Conf. On Intelligent Robots and Systems (IROS '97)*, Vol., pp. 435-440, Grenoble-France

[4] Majid Nili Ahmadabadi, M.E. Barouni, and E. Nakano, " Compliance: Encoded information and behavior in a team of cooperative object handling robots", *in journal of Advanced Robotics, April 2003.*

[5] Aram Zaerpoor, Majid Nili Ahmadabadi, Mohammad R. Barouni, Z. D. Wang, "Extending Constrain-and-Move Strategy to Move Objects on a Desired Path By a Team of Distributed Robots", submitted *in Proceedings of 12th Int. Conf. on Advanced Robotics,(ICAR 2003)*

[6] Donald B.R., Rus D., and Jenning J., "Information invariants for distributed manipulation", *Int. Journal of Robotics Research*, Vol. 16, No. 5, pp. 673-702, October 1997

[7] Y. Yoshikawa "Passive and active closures by constraining mechanisms", *In Proc. of the 1996 IEEE Int. Conf. Robotics and Automation* (ICRA'96), pp. 1477-1484, 1996

[8] D. Rus, B.Donald and J.Jenning," Moving Furniture with Teams of Autonomous Robots", *In Proc, of the 1995 IEEE/RSJ Int. Conf. On Intelligent Robots and Systems" Vol.1, pp.235- 243*

[9] Z.Wang, E. Nakano and T.Matsukawa, "A New Approach to Multiple Robots' Behavior Design for Cooperative Object Manipulation" *In Proc. of 1996 IEEE Int. Symp. On Distributed Autonomous Robotic System (DARS'96), pp.350-361*

[10] Majid Nili Ahmadabadi, M.R. Barouni, and E. Nakano, “A study on rotational lock in a team of distributed object reorienting robots”, *in Proceedings of 10th Int. Conf. on Advanced Robotics,(ICAR 2001)*

[11] Majid Nili Ahmadabadi and Nakano Eiji "Constrain and Move: A new concept to develop distributed object transferring protocols" In Proc. of The 1997 IEEE Int. Conf. On Robotics and Automation (ICRA '97), pp. 2318-23 25, New Mexico-USA

[12] M.Nili Ahmadabadi and E.Nakano, “A Constrain and Move Approach to Distributed Object Manipulation”, *IEEE Transactions on Robotics And Automation, Vol. 17, No. 2, pp.157-171, April 2001*

Bibliographies

NASSER GHARIBAN is Assistant Professor in the Department of Engineering and Technology at Virginia State University. He is also coordinator of manufacturing engineering program in this department. He previously was engineering support director at A. E. Petsche Co. Inc., Arlington Texas. He holds a Ph.D. Degree in mechanical engineering from the University of Texas at Arlington. He involves in manufacturing engineering related research activities including robotics and automation areas.

MOHAMMADREZA BAROUNI-EBRAHIMI is currently Ph.D. candidate in the Department of Computer Science at the University of New Brunswick (Fredericton, Canada). He received a B.Sc. degree in Computer Engineering from Sharif Unviersity of Technology (Tehran, Iran) in 1997 and an M.Sc degree in Machine Intelligence and Robotics from the University of Tehran in 2000.