Asian Journal of Information Technology 5 (9): 938-943, 2006

Motion Planning and Control of Mobile Manipulators

[1]A. Hassam, [1]K. Benmahammed and [1]M. Hamani
Institute of Electronics, University of Ferhat Abbas, Setif 19000, Algeria

Abstract: In this study, motion planning and control of multi-joint mobile manipulators are studied using Genetic Algorithms and Fuzzy Logic. The proposed approach consists of two modules: one for collision free motion planning, the other for fine control of the mobile manipulator. The motion planning module uses a genetic algorithm to generate an optimal path avoiding any collision with possible obstacles existing in robot workspace. The motion control module is a fuzzy controller optimized by a genetic algorithm, which allow to the robot to accurately follow the path generated by the motion planning module without any prior knowledge of the robot dynamics. The effectiveness of the proposed method is evaluated through computer simulation of a two degrees of freedom mobile arm manipulator.

Key words: Genetic algorithms, motion planning, motion control, mobile manipulator

INTRODUCTION

A mobile manipulator generally consists of a robot arm and a mobile platform. The capability of mobile manipulation is important for some autonomous mobile robot applications, which require both a large workspace and dexterous manipulation capability. The mobility of the (mobile) platform substantially increases the performance capabilities of the system. For example, the platform increases the size of the manipulator workspace and enables the manipulator to position the end-effector for executing the task efficiently. One of the unique characteristics of mobile manipulators is the high degree of kinematic redundancy created by the addition of the platform degrees of mobility to the robot manipulator. The redundant Degrees of Freedom (DOF) enable the mobile manipulator to accomplish secondary tasks. For example, joint torques can be optimized, singular configurations of the manipulator can be avoided and decoupled force/position control along the same task direction can be achieved.

Motion planning and control of mobile manipulators are very important in robotics, before any movement and action, the strategy to be followed by the robot must be planned. After knowing the path to be tracked, the immediate problem that the robot faces is to know how to move. Therefore motion control of the robot is required.
Numerous methods on robot motion planning using various approaches are studied[1] proposed an optimal control that searches for the optimal path, but this requires the exact knowledge of the robot dynamic model, which is often very complicated[2,3] proposed a motion planning method for a mobile multi-joint manipulator using genetic algorithm. Several methods for robot motion control based on various approaches are developed[4] proposed an adaptive controller that can compensate partially unknown robot dynamics, but it requires on-line estimation of certain parameters, which is often very complicated and requires a very important computing time. In[5] a fuzzy-sliding mode controller is proposed.

In this study, Fuzzy Logic (FL) and Genetic Algorithm (GA) approaches are used to motion planning and control of mobile manipulators. The proposed approach consists of two modules: one for collision-free motion planning, the other for fine motion control of the mobile manipulator. The motion planning module uses a GA to optimize the path which accurately leads the robot to the goal in minimum time and without any collision with obstacles, the planned path is used as the desired trajectory of the robot motion control module.

The motion control module is a fuzzy controller optimized by a GA, where the GA is used to optimize the knowledge base of the fuzzy controller, this technique is capable of achieving fine robot motion control without any prior knowledge of the robot dynamics.

Motion planning module: Motion planning for a multi-joint mobile manipulator can be divided into two sub problems, one is "path planning", where an optimal path which surly leads the tip of manipulator to goal with minimum time is searched. The other is "collision free sequence generation", where a sequence of variations in joint angles by which any part of the manipulator can avoid collision with obstacles is optimized.

The position of ith joint of manipulator is given by: Fig. 1.

$$Z_i = Z_{i-1} + r_i x e^{j\theta_i} \qquad (1)$$

Corresponding Author: A. Hassam, Institute of Electronics, University of Ferhat Abbas Setif 19000, Algeria

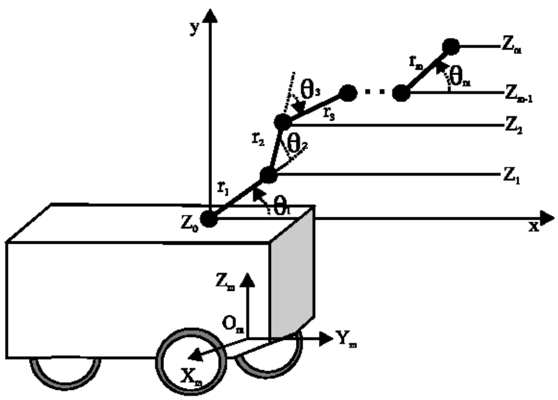

Fig. 1: Joints positions in cartesian space

Joints \ Time-steps	1	2	3	....	n
	-13	-11	-8		2
	6	23	-13		19
	-25	17	-8		0
	11	0	-7		-29

Fig. 2: Structure of chromosome

With $\quad Z_i = x_i + j \times y_i \quad$ and $j^2 = -1$

Where r_i is the length of ith link and θ_i is the angle of ith joint. The space occupied the segment between Z_i and Z_{i-1} is represented by:

$$D_i = \{t \times Z_i + (1-t) \times Z_{i-1};\ 0 \leq t \leq 1\} \qquad (2)$$

where t is a parameter. Then, the space where the manipulator exists is given by:

$$D_t = D_1 \cup D_2 \cup \ldots \cup Dm = \bigcup_{i=1}^{m} D_i \qquad (3)$$

Where m is the number of joints If

$$D_t \subset D_f \qquad (4)$$

where D_f is free-space, the space where there is no obstacles, is satisfied the manipulator never collide with obstacles. Here, θ is defined as $\theta = (\theta_1, \theta_2, \theta_3 \ldots \theta_m)T$, if θ^* which satisfies the relation of (1, 2, 3 et 4 at all time-steps can be found, the collision-free sequence of movement can be realized. Genetic algorithms are very powerful procedures of optimization, they are less likely to be trapped in local minima because of the parallel exploration of a whole of solutions to a given problem[6].

Genetic operations used in our method are explained as follows[3]:

Genetic coding: Variations in joint angle are coded as gene and the chromosome has two dimensional structures Fig. 2. Each number describes a variation in the angle of each joint at a time step. The length of chromosome is variable so that chromosome can represent manipulator movement which has any time length.

Fitness: The purpose of GA is to generate an optimal solution satisfying the minimization of the cost function given by:

$$J = c1 \times Lc + c2 \times \sum_{i=1}^{m} \sum_{k=2}^{n} |\theta i(k) - \theta i(k-1)|$$

Where:

$$\sum_{i=1}^{m} \sum_{k=2}^{n} |\theta i(k) - \theta i(k-1)|$$

represents the sum of variations in joint angles, its minimization allows to obtaining a smooth trajectory. Lc represents the length of the chromosome; its minimization gives short trajectory, therefore a minimum execution time. The weighting coefficients c1 and c2 are selected empirically to avoid overflows.

Crossover: Crossover operation used in our method is called analogous crossover because the crossover point is decided by the function of phenotype and not by the position of genotype. The procedure of analogous crossover used in our method is as follows:

- Choose two individuals as parent (Parent A and Parent B) at random.
- Choose crossover point of parent A randomly.
- Calculate the position of the tip of parent A at the chosen crossover point.
- Compare the position of the tip at each step of parent B with calculated position of parent A.
- Select a crossover point of parent B which is the nearest to the crossover point of parent A.
- Carry out the partial exchange of genes between the two parents A and B.

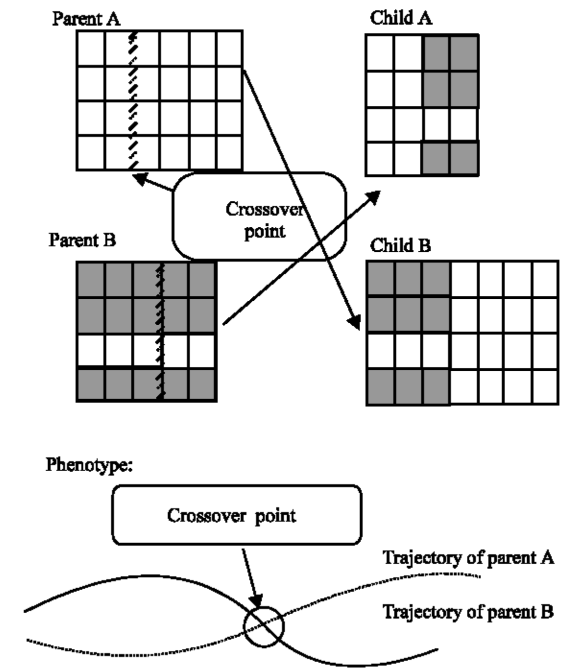

Fig. 3: Analogous crossover

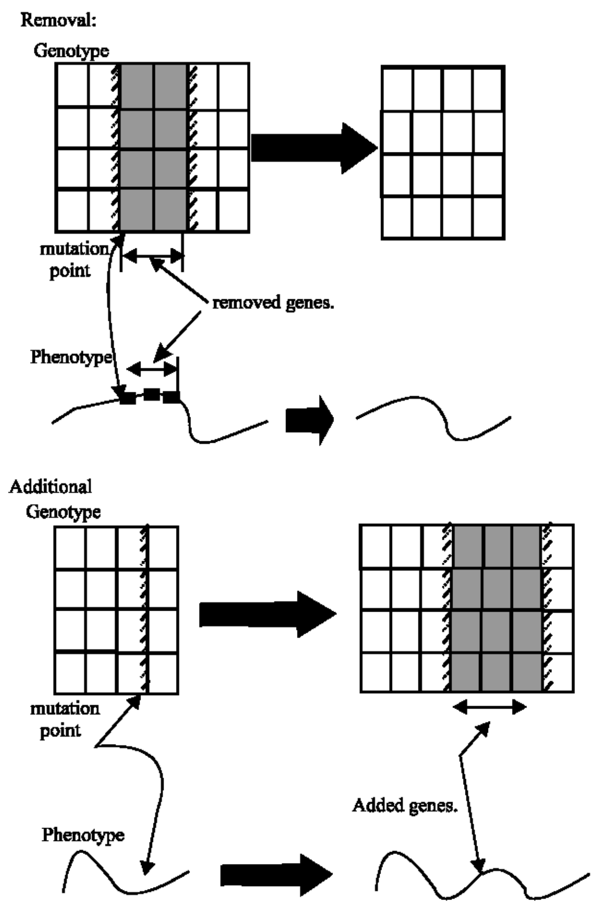

Fig. 4: Mutation (removal and additional)

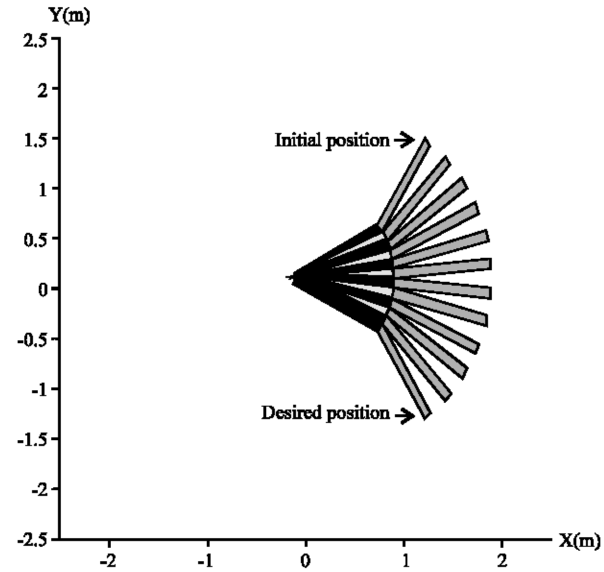

Fig. 5: Planed path without obstacles

Mutation (Addition): This operation adds random length of gene at random point to the chromosome Fig. 4. By this operation, new trajectory is added to individual in phenotype.

Mutation (Removal): Mutation removal operation removes random length of gene from chromosome at random point Fig. 4. Using this operation, sequence of movement is changed partly then individuals will have more variety.

Selection: this operation allows to individuals with better qualities to often take part in the generation of next population that those of less qualities.

Constraint: After selection, crossover and mutation operations, any individual child does not satisfy the relations (1-4) will be replaced by one of his parents.

Computer simulation: To illustrate the efficiency of robot motion planning module, the proposed method is applied to motion planning for two degrees of freedom planar manipulator with link lengths $l_1 = 1$m, $l_2 = 1$m. The initial robot configuration is at $(\theta_1, \theta_2) = (30°, 30°)$, where θ_1 and θ_2 are the joint angles. The initial tip position of the second link is at (1.366, 1.366) in Cartesian space and the target is at (1.366,-1.366) Fig. 5. Genetic algorithm parameters used in this application are:

- Chromosome dimensions are m = 2, is the number of joint, n is variable <150 represents time-steps.
- Weighting coefficients c1 and c2 are equal to 10^{-5}, 10^{-3}, respectively.

Fig. 6: Planed path with obstacles

The task is to plan an optimal path that leads the tip of manipulator to the closest target while any part of manipulator can avoid collision with obstacles.

Simulation results: First, no obstacles are introduced into the robot workspace Fig. 5, we note that the proposed algorithm managed to move the tip of manipulator from initial position to the desired position while following the shortest path. There after, two obstacles whose positions are at (0.8,0) and (- 1.5, 0), are introduced into the robot workspace Fig. 6. As it can be viewed the proposed algorithm managed to move the tip of manipulator from initial position to the desired position without any collision with obstacles.

Motion control module: The dynamics of a mobile manipulator with m degrees of freedom can be modeled in matrix form as follows:[7]

$$M(q)q + C(q,q) + G(q) = \tau,$$

Where:

q:	is the articular variable.
M(q):	is the inertial matrix of the robot manipulator.
C(q,q) :	is the vector of the centrifugal forces.
G(q) :	is the gravitational vector.
τ:	is the actuator torque vector.

The controller design problem can be described as: given a desired manipulator joint position $q_d(t)$, design a control law for the actuator torque τ, which drive the robot to move, such that the manipulator joint position

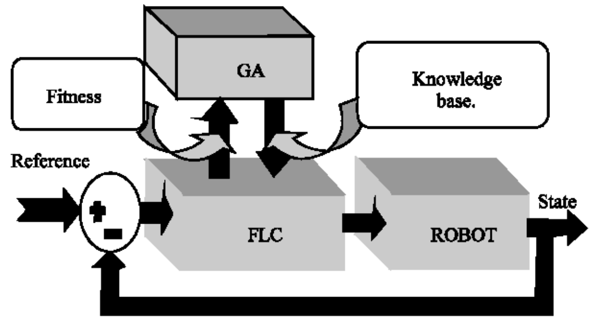

Fig. 7: Genetic structure of FLC

q(t) tracks $q_d(t)$. There are many tasks that have to be performed in order to design a FL Controller (FLC) to control a concrete system. It is known that the derivation of the Knowledge Base (KB) is the only one directly depending on the controlled system and it presents a significative importance in the design process. The more used method in order to perform this task is based directly on extracting the expert experience from human process operators. The problem arises when these are not able to express their knowledge in terms of fuzzy control rules. In order to avoid this drawback, researches have been investigating automatic learning methods for designing FLC by deriving automatically an appropriate KB for the controller system without necessity of its human operator.

The GAs have demonstrated to be powerful tool for automating the definition of KB, this advantage have extended the use of GAs in the development of a wide range of approaches for designing FLC in the last years, one of these approaches is studied in this study. The proposed approach is very specific since a GA is used to automatically deriving the FLC knowledge base, this approach is designed for applying in two-inputs-one-output systems. A schematic diagram of the proposed controller is shown in Fig. 7.

Fuzzy logic controller description: The proposed controller has two inputs and one output where the inputs are the error and the variation in error on the articular position and the output is the actuator torque. The membership functions used for partitioning inputs and output universes of discourse are triangular and symmetrical.

We associated to the variable error (x1) three terms: Positive (P), Zero (Z) and Negative (N) distributed on the universe of discourse [-0.05, 0.05] and for the variation in error (x2) and actuator torque (y), five terms are associated: Negative Large (NL), Negative Small (NS), Zero (Z), Positive Small (PS) and Positive Large (PL) distributed on [-0.06, 0.06] and [-5, 5], respectively. This controller is based on the (max-min) MAMDANI

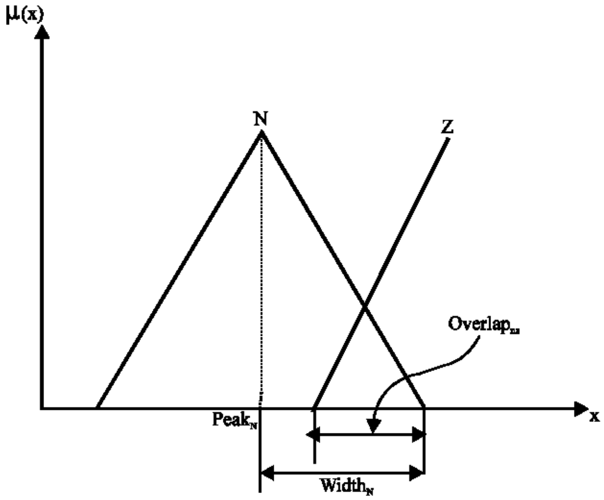

Fig. 8: Parameters describing the membership function

implication rule, the fuzzy conjunction (min) and by the center-of-area method for the defuzzification.

Genetic algorithm description: Genetic operations used is this study are:

Genetic coding: coding used in our method is a multi-parameterized and concatenated, where several parameters are coded and juxtaposed the ones after the others. The parameters in our case are the fuzzy rules and the parameters describing the FLC membership functions. As the membership functions are triangular and symmetrical, two parameters only are necessary to describe them, these parameters can be the peak, the base (width) or the overlap Fig. 8. To avoid having big length of chromosome, we considered it useful to fix certain parameters:

- For the variable of error, the fixed parameter is $peak_Z$.
- For the variation in error and the actuator torque, the fixed parameters are $peak_{NL}$, $peak_Z$ and $peak_{PL}$.

Therefore, parameters to be optimized are:

- $peak_N$, $peak_P$, $width_N$, $width_Z$ and $width_P$ for the variable of error.

- $peak_{NS}$, $peak_{PS}$, $width_{NL}$, $width_{NS}$, $width_Z$ $width_{PS}$ and $width_{PL}$ for the variation in error and the actuator torque.

That gives in total 19 parameters to be optimized, each parameter is coded on 5 bits. The length of chromosome required to code the 15 fuzzy rules (rule base) and the 19 parameters describing the membership functions (data base) is (19 x 5+15) = 110 alleles. Since it is about the control of a two degrees of freedom manipulator, we need one fuzzy controller for each one of its axes. The optimization of the two FLCs knowledge bases is carried out simultaneously by coding them on the same chromosome, then the length of the chromosome becomes 110x2 = 220 alleles.

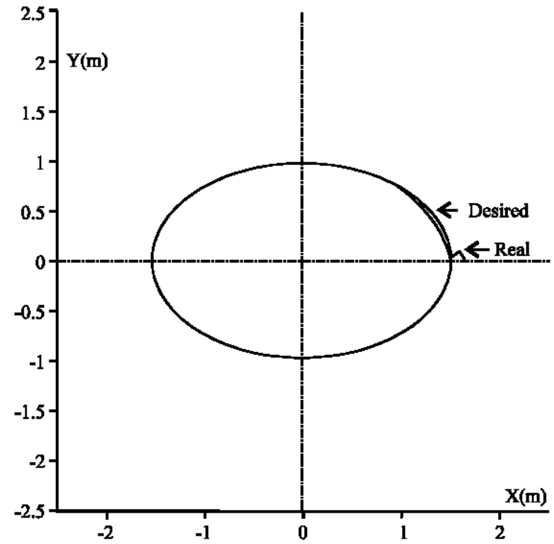

Fig. 9: Tracking performances

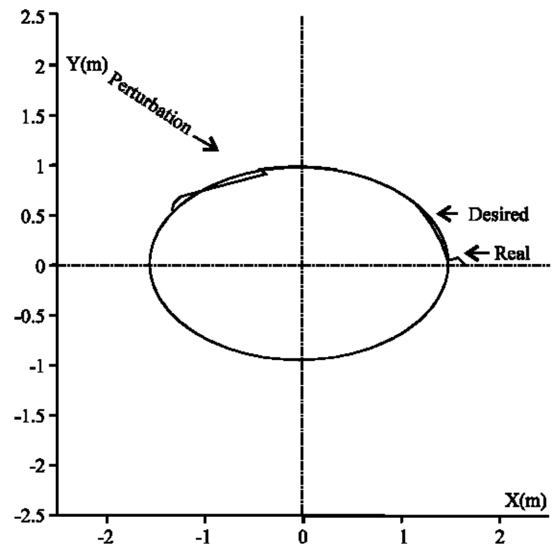

Fig. 10: Tracking performances under severe conditions

Fitness: The problem of this controller returns to a least square problem, where we want that the controller minimizes the quadratic error while avoiding abrupt changes in the control action, this is why our objective function is given by:

$$J = \sum_{K=1}^{300} \sqrt{e^2(k) + \dot{e}^2(k)}$$

Genetic operators: The proposed GA employs a roulette wheel selection with replacement, a standard two-point crossover operator and a mutation operator which has the same principle as the standard mutation operator, except if the mutation point is in the part coding the fuzzy rules, we change the value of the allele by a random value between (1 and 5) different from its current value.

Numerical simulation: The proposed controller is applied to control the movement of a two degrees of freedom manipulator so that the tip of this robot can track the elliptic trajectory represented in Cartesian space by Fig. 9. GA parameters used in this application are:

- Crossover probability is fixed at 0.7.
- Population size and the maximum number of generation are 100 and 500, respectively.
- Mutation probability is fixed at 0.03.

Simulation results: Fig. 9 shows that the tracking of the desired trajectory is completed successfully, which justifies the effectiveness of the proposed controller. To test the robustness of the proposed controller vis-a-vis the parametric variations and with external disturbances, we abruptly increased the mass of the second segment of 50% of its initial mass at (t = 1s). In this study, simulation results are represented in Fig.10, which shows that the proposed genetic FLC is robust vis-a-vis the parametric variations and the external disturbances.

CONCLUSION

In this study, we proposed a method for motion planning and control of mobile manipulators based on two original approaches: Fuzzy control and genetic algorithms. The studied approach is divided into two modules, one for motion planning and the other for motion control of the mobile manipulator. The motion planning module uses a genetic algorithm to optimize the path to be tracked to move the manipulator tip from the initial position to the desired position in minimum time and without collision with obstacles. Genetic operations used in this module are very effective, particularly mutation (removal) which reduces effectively the length of the chromosome, therefore the path to be tracked. We confirm the proposed method can be applied in various workspaces having different arrangements of obstacles. The planned path is used as the desired trajectory of the robot motion control module. The motion control module uses a fuzzy controller optimized by a genetic algorithm, where GA is used to develop the knowledge bases of the fuzzy controllers associated to the mobile manipulator articulations. The use of fuzzy control eliminated the need for an exact mathematical model and the use of the genetic algorithm eliminated the need of a human expert in the design phase for such controller. The study considered here shows the effectiveness of designing fuzzy controllers by genetic algorithms to achieve a fast, precise and robust tracking control of the robot manipulators and especially without commutation effects. For this mobile manipulator, we have treated only the "arm robotized" in autonomous mode. Indeed, the control of the arm alone is sufficient to illustrate the specific problems of supervision to our architecture of control and allow, moreover, to limit the complexity of the control to be realized. Later on, it will be possible to consider the robot in its entirety by developing controls of the arm in autonomous mode (work already completed) and those of the mobile platform (initiated work) and both coupled (work under development). But initially, only the control of the arm manipulator in autonomous mode was developed.

REFERENCES

1. Jindong, T., Ning Xi and Yuechao Wang, 2003. Integrated task planning and control for mobile manipulators. The Intl. J. Robotics Res., 22: 337-354.
2. Yanrong, H. and Simon X. Yang, 2004. A knowledge based genetic algorithm for path planning of a mobile robot. Proc. IEEE Int'l. Conf. On Robotics and Automation, pp: 4350-4355.
3. Nishimura, T., K. Sugawara, I. Yoshihara and K. Abe, 1999. A motion planning method for a hyper multi-joint manipulator using genetic algorithm. Proc. IEEE Int'l. Conf. on Systems, Man and Cybernetics, 4: 645-650.
4. Bruno Siciliano and Luigi Villani, 2001. Two-time scale force and position control of flexible manipulator. IEEE Int'l. Conf. on robotics and Automation, pp: 2729-2734.
5. Seok Jo Go, Min Cheol Lee and Min Kyu Park, 2001. Fuzzy-sliding mode of automatic polishing robot system with the self tuning fuzzy inference based on genetic algorithm. Proc. IEEE Int'l. Conf. on robotics and Automation, pp: 2962-2967.
6. Winter, G., J. Periaux, M. Galan and P. Cuesta, 1995. Genetic Algorithms in Engineering and Computer Science. British Library Cataloguing in Publication data.
7. Fu, K.S., R.C. Gonzalez and C.S.G. Lee, 1987. Robotics: Control, Sensing, Vision and Intelligence. Mc Gram-Hill Book Company, 1987.