ROBOTIC MANIPULATOR PATH PLANNING BY GENETIC ALGORITHMS

Luiz Eduardo Nicolini do Patrocínio Nunes, luiz@unitau.br
Victor Orlando Gamarra Rosado, victor@feg.unesp.br
Universidade de Taubaté, Rua Daniel Danelli, s/n, Taubaté – SP, CEP 12060-440
Universidade Estadual Paulista, Av. Dr Alberto P. da Cunha, Guaratinguetá – SP, CEP 12516-410

Francisco José Grandinetti, grandi@unitau.br
Universidade de Taubaté, Rua Daniel Danelli, s/n, Taubaté – SP, Cep 12060-440

Abstract. *The path planning of robotic manipulators consists of finding continuous movements that take the arm of one given initial configuration until a desired position in the work space. Diverse works available in the literature shown the genetic algorithms application path planning in robotics. Many of these works present complex techniques in the genetic algorithm implementation, involving the changeable size chromosomes manipulation. The inverse kinematics solution by genetic algorithms also has been proposed in the literature, however, for isolated points. This work considers the application of a genetic algorithm based in the direct kinematics calculation, for generate trajectories to a robotic manipulator with three degrees of freedom. The implemented algorithm calculates the inverse kinematics for all the trajectory's points. In this work, the positioning error and the joints angular displacement are boarded through a multi-objective function. The obtained results show the efficiency of the used methodology.*

Keywords*: Trajectory, genetic algorithm, inverse kinematics, robotic manipulator*

1. INTRODUCTION

The inverse kinematics determines the angles of the junctions that result in the desired position of the terminal organ of a manipulator with relation to the reference coordinate system. The solution of the inverse kinematics is difficult whereas the mapping between the Cartesian space and the junction space is non-linear and evolve equations that can have multiple solutions (Craig, 1989). Several methods for the solution of the inverse kinematics, based on Genetic Algorithms (GAs) have been proposed in the literature.

In Kalra et al. (2003) the inverse kinematics problem using genetic algorithms was explored for isolated points. In that work, the algorithm supplied the two possible solutions to the inverse kinematics problem, and it can be chosen, as better solution, which one presented bigger fitness value.

The inverse kinematics problem for isolated points was also elaborated by Parker et. al. (1989). The objective of that work was to place the performer in the correct position and to minimize the displacements of the manipulator junctions.

Eydgahi and Ganesan (1998) presented the application of genetic algorithms for the generation and adjustment of the pertinence functions of diffuse groups for the solution of the inverse kinematics of robotic manipulators. The presented method converges more quickly to the solution in comparison to methods based only on diffuse systems or based on Artificial Intelligence.

Buckley et al. (1997) used a genetic algorithm in the solution of the problem of the inverse kinematics of a robotic manipulator with high kinematics redundance. As the first results had not presented acceptable solutions, the genetic algorithm was adjusted to draw out the diversity of the population and to reduce the space of search for the application of the kinematic knowledge of the junction 2.

The planning of trajectories for robotic manipulators consists in finding continuous movements that take the arm of one given initial configuration until a desired position in the work space (Pires and Machado, 1999).

Several available works in the literature have shown the application of genetic algorithms in the generation of trajectories for robotic manipulators.

Toogood et al. (1995) used a genetic algorithm for the attainment of a free trajectory of collisions for a robotic manipulator of three degrees of freedom (DOF) in a space containing fixed and known obstacles. Besides avoiding collisions, the trajectory could be optimized for smaller distance, minor time or minimum torques.

Tian and Collins (2003, 2004) proposed a genetic algorithm using real codification for the search of an excellent trajectory of a redundant manipulator. The evaluation function was based on multiple criteria such as total displacement of all the junctions and uniform speed in the Cartesian spaces and junction. For validation of this approach, simulations were carried out in a workspace with and without obstacles.

Santos et al. (2005) presented a genetic algorithm capable to solve the problem of the inverse kinematics for all the trajectories points. They used the Reduction Technique of Search Space (RTSS), that, according to the authors, improves as much the speed as convergence accuracy avoiding leaps between multiple solutions.

Some works present complex techniques in the implementation of the genetic algorithm for the calculation of trajectories of robotic manipulators. Such techniques involve the chromosomes manipulation of variable size (Marques *et al.*, 1996, Davis, 1996).

The objective of this work is to present the application of a genetic algorithm for the generation of trajectories of a robotic manipulator planar with three degrees of freedom. In the implemented algorithm, an individual chromosome represents the angles of the manipulator junctions for a determined point. By presenting minor computing cost, the angles of the junctions were codified using the real representation instead of the binary one (Tian and Collins, 2003). The evaluation function (fitness) has multi-objective character and it is defined on the basis of two criteria: minimum angular displacement of the end-effector in the Cartesian space and minimum angular displacement of the manipulator junctions, using the ponderation of the objective method. The genetic algorithm calculates the inverse kinematics for all the points that describe a linear trajectory.

2. PROBLEM FORMULATION

In this work was considered a robotic manipulator planar with three degrees of freedom (Fig. 1). The joint angles θ_1, θ_2 and θ_3 can vary between $-\pi$ and π. The lengths of the links l_1, l_2 and l_3 are 25 cm, 15 cm and 10 cm respectively. The end-effector must follow linear trajectories starting in the Cartesian coordinate (12.0004, 14.9995) and finishing in different final points.

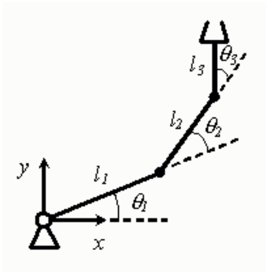


Figure 1. Robotic Manipulator Planar

The initial configuration corresponds to the vector of variable of the junctions $\{0.0307\ 1.8756\ 1.5691\}^T$ in radians, as illustrates the Fig. 2. In the Fig. 2, the area between the circumferences of radius R_1 and R_2 corresponds to workspace of the manipulator, being $R_1 = l_1 + l_2 + l_3$ e $R_2 = (l_2^2 + (l_1 - l_3)^2)^{0.5}$. The final configurations are the coordinates (x, y) of the end-effector, gotten through the Eq. (1) with the current angles generated by the GA.

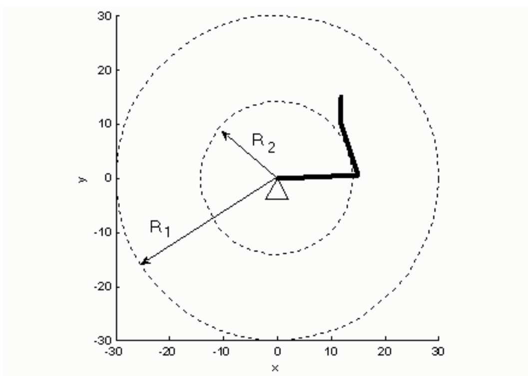


Figure 2. Initial configuration of the manipulator

The final configurations are the coordinates (x, y) of the end-effector, gotten through the Eq. (1) with the current angles generated by the GA.

$$\begin{Bmatrix} x \\ y \end{Bmatrix} = \begin{Bmatrix} l_1\cos(\theta_1) + l_2\cos(\theta_1+\theta_2) + l_3\cos(\theta_1+\theta_2+\theta_2) \\ l_1\sin(\theta_1) + l_2\sin(\theta_1+\theta_2) + l_3\sin(\theta_1+\theta_2+\theta_2) \end{Bmatrix} \tag{1}$$

The trajectory of a manipulator consists in a group of strings that represent the positions of the junctions between the initial and final robot configurations. A Genetic Algorithm was adopted to look for an optimal global path for the manipulator.

3. GENETIC ALGORITHMS

The Genetic Algorithms (GAs) constitute one technique of search and optimization, highly parallel, inspired in the evolution principle of Darwin. The natural principles, in which the GAs were inspired, are simple. The selection principle privileges the most apt individuals and, therefore, with more probability of reproduction. The individuals with more descendants have more chances of transmitting their genetic codes to the next generations (Michalewicz, 1994). Such genetic codes constitute the identity of each individual and are represented in the chromosomes. These principles are emulated in the construction of computing algorithms that search the best solution for one determined problem, through the evolution of populations of codified solutions through artificial chromosomes. The components of a GA include: initialization, selection, crossing and mutation according to illustrates Fig. 3 (Kalra *et al.*, 2003):

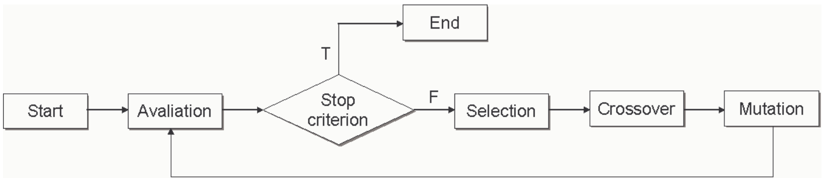


Figure 3. Components of a GA

3.1. Individual representation

The success of a genetic algorithm for a specific problem of optimization depends on the representation of an individual in the population (kalra *et al.*, 2003). Each possible solution in the search space is represented by a sequence of symbols *s* generated from an alphabet (binary or real). Each *s* sequence corresponds to a chromosome and each element of *s* is equivalent to a gene.

For a robotic manipulator, the individuals in a population can be represented, with real codification, through the joint angles: $\{\theta_1\ \theta_2\ \theta_3\}$. The real codification was chosen to avoid succeeding conversions of the binary code or gray for real values, saving, thus, computing time.

3.1. Initialization

In the initialization process, a population of chromosomes is generated randomly. The size of the population affects the efficiency and the performance of the GA (Goldberg, 1989). A population of small dimension can take the GA to converge quickly to a maximum local, while a very big population, damages the computing performance of the algorithm. The initial population for a robot with three degrees of freedom is generated randomly respecting the inferior (*L*) and superior (*U*) limits of each variable of junction:

$$\theta_i^L \leq \theta_i \leq \theta_i^U \qquad i = 1,2,3 \tag{2}$$

3.3. Evaluation

To each structure (solution) is associated a numerical value (fitness) that represents the quality of this structure and indicates its aptitude degree. The value of fitness is gotten through the objective function. The objective function of this study aims at the minimization of the position error of the end-effector manipulator and the smallest joint angular displacement. The positioning error is calculated through the Euclidean Distance between the current and final coordinates of the manipulator:

$$E_p = \sqrt{(x_i - x_f)^2 + (y_i - y_f)^2} \tag{3}$$

Being (x_f, y_f) the desired position and (x_i, y_i) the current position, gotten through the Direct Kinematics calculation of the manipulator (Eq. 1) with the use of the current angles generated by the GA. The angular error is also gotten through the Euclidean Distance between the initial and final configurations of the joint angles of the manipulator:

$$E_a = \sum_{i=1}^{3} \left\| \{\theta_{i,f}\} - \{\theta_{i,in}\} \right\| \tag{4}$$

Where $\{\theta_{i,in}\}$ are the initial configuration angles of the manipulator, $\{\theta_{i,f}\}$ are the current angles generated by the GA and ||.|| denotes the Euclidean distance. In this work, the positioning error and the angular displacement are approached together through a multi-objective function (Nunes *et al.*, 2006). Using the weighting factors method, that satisfies the restriction $\omega_1 + \omega_2 = 1$, the optimization problem is defined as the inverse of the error:

$$fitness = \frac{1}{\omega_1 E_p + \omega_2 E_a} \tag{5}$$

3.4. Seletion

The selection process in GAs chooses individuals for the reproduction. The selection is based on the individuals aptitude: apter individuals have more probability of being chosen for the reproduction. The selection method chosen for this work was Stochastic Universal Sampling (SUS), in which the individuals are mapped out in adjacent segments whose length is the same as the value given for the evaluation function to each individual. In this method it is used N hands equally spaced between them (N = number of parents to be selected) and the roulette spins only once. The chosen parents are the individuals marked for the N hands. The distance between the hands will be $1/N$ and the position of the first hand is given for a number generated randomly between 0 and $1/N$. The method SUS is considered fast enough for the serial processing and more efficient than the Roulette selection methods, Stochastic Rest and Ranking (Baker, 1987).

3.5. Crossing and mutation

The individuals selected for the following population are recombined through the Crossover operator. This operator is considered the main characteristic of the GAs. The pairs of individuals are chosen randomly and new individuals are created from the interchange of the genetic material. The descendants will be different, however, with genetic characteristics of both. This method (single-point crossover) is the most applied one and was used in this work. The chromosomes created from the crossover operator are, later, submitted to the mutation operation. Based on the probability pm of mutation, the content of a chromosome position is modified.

5. RESULTS

Before the calculation of all the points of the linear trajectory between the initial point and the final coordinate, the Genetic Algorithm was applied in the inverse kinematics solution for isolated points, with objective of being determined the ideal values of the ponderation factors (ω_1 and ω_2) of the multi-objective function given for the Eq. (5). After several tests, it was adopted the values of 0.12 and 0.88 for ω_1 and ω_2 respectively, with priority for the minimum angular displacement.

5.1. Trajectories generation

The described GA in this work was used to get a linear trajectory of an initial point to an final point. For each point (p_1, p_2, ..., p_n) of the trajectory, the GA is executed, in an iterative process, until the limit of 500 generations be reached. After the point p_i is found, its values are brought up to date as initial point to find the point p_i+1 (Nunes et. al., 2006). The simulated trajectories in this work have 120 points. To facilitate the identification of the simulated trajectories, they will be cited by numbers, according to the Tab. 1. All the simulations were carried out from the same initial configuration of the manipulator, reaching different final points.

Table 1. Identification of the Simulations

Simulation nº.	**Final Point**
1	(20, 10)
2	(-10, 20)
3	(26,5, 5,5)
4	(-20, 20)

The following parameters of control were used for the Genetic Algorithm: population size = 90 individuals, crossing probability = 0.7 and mutation probability = 0.09.

In relation to the simulation 1, the Fig. 4 shows an amplified region of the reference trajectory to be followed by the manipulator from its initial position until the final point (20, 10). The trajectory generated by the GA, inspite of being very near to the reference trajectory, presents acceptable small deviations in the task.

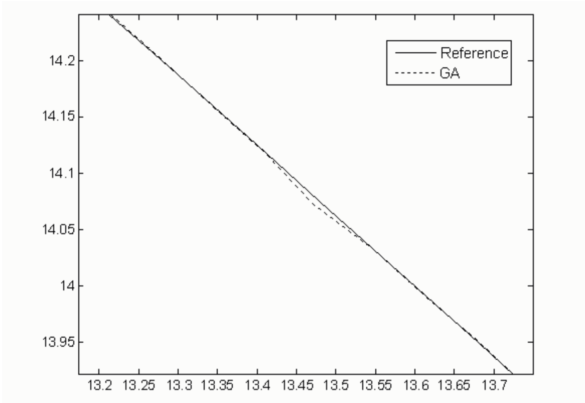


Figure 4. Trajectory gotten through GA

The Figure 5 shows the distances between each point of the reference trajectory and the correspondent points of the trajectory generated by the genetic algorithm. In this simulation, the biggest deviation in relation to the reference trajectory was 0.0056 cm.

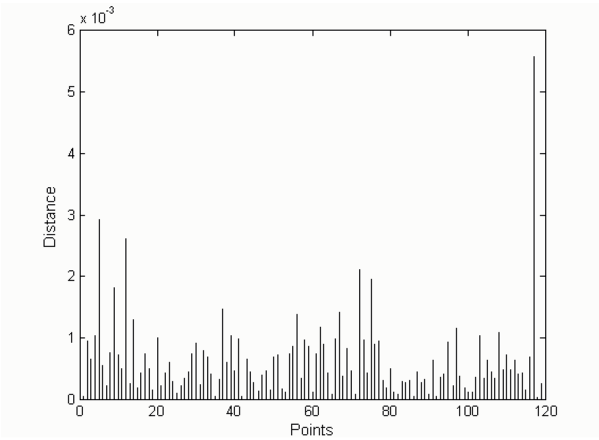


Figure 5. Distances between the trajectory points

To permit the manipulator covers the reference trajectory with minimum angular displacements, calculation of the angular errors was adopted in the Eq. (5), these errors were added in the fitness function. The Figure 6 illustrates the succeeding configurations of the manipulator between the initial point and the final point.

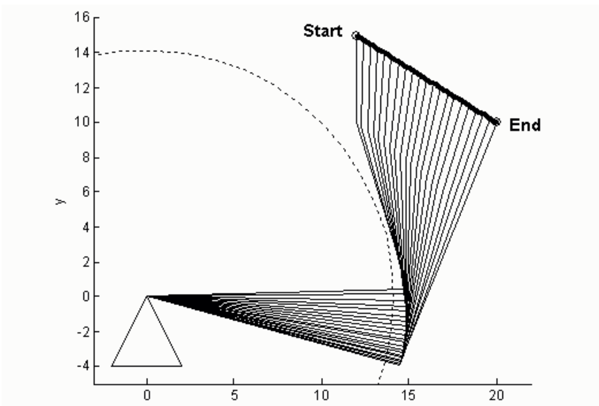
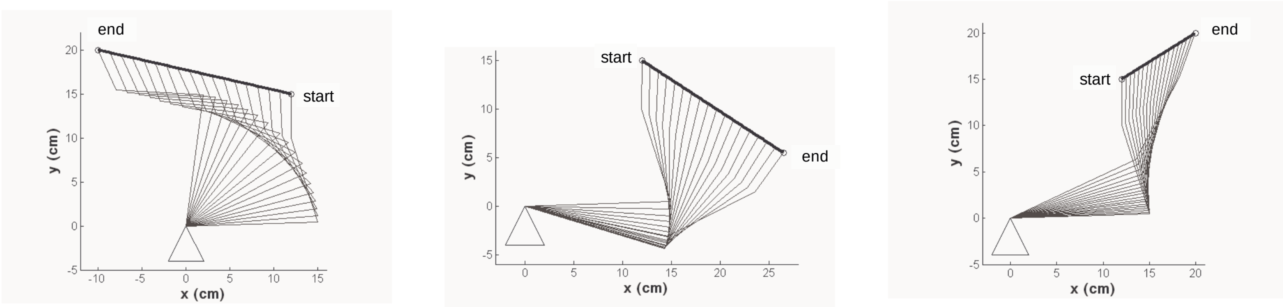


Figure 6. Succeeding Configurations of simulation 1

The results of the subsequent simulations are shown in the Figure 7 (a), (b) and (c), that correspond to the simulations from 2 to 4 respectively. These figures show the succeeding configurations of the manipulator from the initial point until the final point of each trajectory. As it can be observed in these figures, the movement of the junction angles of the manipulator is relatively soft and the trajectories gotten through the genetic algorithm do not present big leaps or deviations in relation to the reference trajectories.



(a) (b) (c)

Figure 7. Succeeding configurations of simulations 2, 3 and 4

The maximum deviations of all the trajectories generated in each simulation for the genetic algorithm, in relation to the reference trajectories, are presented in the Table 2. With exception of the simulation 4, which presented a maximum deviation of 0.0558 cm, all the simulations presented small deviations.

Table 2. Deviations in Relation to the Trajectory of Reference

Simulation nº.	**Final Point (x,y)**	**Bigger Deviation (cm)**
1	(20, 10)	0.0056
2	(-10, 20)	0.0172
3	(26.5, 5,5)	0.0324
4	(20, 20)	0.0558

The angles generated by the Genetic Algorithm for the final points of each trajectory were substituted in the Eq. (1) in order to get the positions x and y of the manipulator in those points. The gotten results are presented in Table 3, that show the biggest relative error of position was 0.1196% in x and 0.12% in y, relative errors referring to the simulation 3. As it can be observed in Tab. 3, inspite of the Genetic Algorithm having as priority, in this work, the total displacement minimization of the junction angles of the manipulator, it was gotten small positioning errors.

Table 3. Relative Errors of Position

Simulation nº.	**Final Point (x, y)**	**Actual Position (x, y)**	**Relative Error in x (%)**	**Relative Error in y (%)**
1	(20, 10)	(20.0000, 9.9999)	0	0.001
2	(-10, 20)	(-9.9993, 20.0001)	0.007	0.0005
3	(26.5, 5.5)	(26.4683, 5.5066)	0.1196	0.12
4	(20, 20)	(19.9955, 19.9971)	0.0225	0.0145

The results were satisfactory, whereas the robot reached the desired positions without big oscillations in the generated trajectories, with soft displacement of the junction angles.

6. CONCLUSION

A genetic algorithm, based on the direct kinematics calculation, was implemented for the generation of linear trajectories for a robotic manipulator with three degrees of freedom. Whereas the GA uses the direct kinematics simply, singularities do not constitute problems. The evaluation function (fitness) had multi-objective character and was defined based on two criteria: minimum displacement of end-effector in the Cartesian space and minimum angular displacement of the manipulator junctions, using the ponderation method of the objectives. According to the gotten results, the movement of the angles of the manipulator junctions was relatively soft, and the trajectories gotten for the genetic algorithm did not present big leaps or deviations in relation to the reference trajectories. The implemented genetic algorithm had, as priority in this work, the total displacement minimization of the junction angles of the manipulator, despite this restriction, it was gotten very small errors of positioning in the final points of each trajectory. Depending on the type of application, the results can be considered satisfactory, whereas the robot reached the desired positions without big oscillations in the trajectories generated, with soft displacement of the junction angles.

REFERENCES

Ata, A. A., Myo, R. T., 2005, "Optimal Point-to-Point Trajectory Tracking of Redundant Manipulators Using Generalized Pattern Search", International Journal of Advanced Robotic Systems, Vol. 2, N. 3, pp. 239-244.

Baker, J., 1987, "Reducing bias and inefficiency in the selection algorithm", In Genetic Algorithms and their Applications: Proceedings of the Second International Conference on Genetic Algorithms, pp. 14–21.

Buckley, K. A., Hopkins, S. H., Turton, B. C. H., 1997, "Solution of Inverse Kinematics Problems of a Highly Kinematically Redundant Manipulator Using Genetic Algorithms", 2nd Int. Conf. on Genetic Algorithms on Engineering Systems: Innovations and Applications, 2-4 Sep., pp. 264-269.

Craig, J, J., 1989, "Introduction to Robotics: mechanics and control", Addison-Wesley Publishing, 2nd ed., Mass, USA.

Davis, L., 1996, "Handbook of Genetic Algorithms", International Thomson Computer Press, Boston, MA, USA, pp. 144-165.

Eydgahi, A. M., Ganesan, S., 1998, "Genetic Based Fuzzy Models for Inverse Kinematics Solution of Robotic Manipulators", IEEE International Conference on Systems, Man and Cybernetics, Vol. 3, pp. 2196-2201.

Goldberg, D. E., 1989, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley Publishing, Mass, USA.

Kalra, P., Mahapatra, P. B., Aggarwal, D. K, 2003, "On the Solution of Multimodal Robot Inverse Kinematics Functions using Real-coded Genetic Algorithms", IEEE International Conference on Systems, Man and Cybernetics, Vol. 2, pp. 1979-1984.

Marques, A., Sequeira, J., Ramalho, M., Gonçalves, R., 1996, "Planeamento de Trajectórias de um Manipulador Robótico usando Algoritmos Genéticos", Proceedings of 1st workshop on Genetic Algorithms and Artificial Life - AGVA96, pp. 26-30.

Michalewicz, Z., 1994, "Genetic Algorithms + Data Structures = Evolution Programs", Springer-Verlag, 3 ed.

Nunes, L. E. N. P., Rosado, V. O. G., Grandinetti, F. J., 2006, "Geração de Trajetória de um Manipulador Robótico Planar de Três Graus de Liberdade Através de Algoritmos Genéticos", XVI CBA – Congresso Brasileiro de Automática, Out. 3-6, Salvador – BA.

Parker, J. K., Khoogar, A. R., Goldberg, D. E, 1989, "Inverse Kinematics of Redundant Robots using Genetic Algorithms", Proceedings of IEEE International Conference on Robotics and Automation, Vol. 1, pp. 271-276.

Pires, E. J. S.; Machado, J. A. T., 1999, "A Trajectory Planner for Manipulators using Genetic Algorithms", Proceedings of IEEE International Symposium on Assembly and Task Planning, pp. 163-168.

Tian, L., Collins, C., 2003, "Motion Planning for Redundant Manipulators Using a Floating Point Genetic Algorithm", Journal of Intelligent and Robotic Systems, n. 38, pp. 297-312.

Tian, L., Collins, C., 2004, "An Effective Robot Trajectory Planning Method Using a Genetic Algorithm", Elsevier Mechatronics, n. 14, pp. 455-470.

Toogood, R., Hao, H., Wong, C., 1995, "Robot Path Planning Using Genetic Algorithms", Proceedings of IEEE Systems, Man and Cybernetics Conference, Vol. 1, pp. 489-494.

Santos, A. F., Junior, M. G., Lopes, H. S., 2005, "Cinemática Inversa de Trajetórias de Manipuladores Robóticos Redundantes Utilizando Algoritmos Genéticos com Redução Progressiva do Espaço de Busca", VII SBAI - Simpósio Brasileiro de Automação Inteligente, São Luís, MA, Brasil.