CONTROLO 2008
8th Portuguese Conference on Automatic Control
University of Trás-os-Montes and Alto Douro, Vila Real, Portugal
July 21-23, 2008

# ROBOTIC SYNTHESIS USING A HIERARCHICAL MULTI-OBJECTIVE EVOLUTIONARY ALGORITHM

**E. J. Solteiro Pires** * **P. B. Moura Oliveira** *
**J. A. Tenreiro Machado** **

* *Dep. Eng. Electrotécnica, UTAD University,CITAB*
*5000-911 Vila Real, Portugal*
{epires,oliveira}@utad.pt
** *Dep. Eng. Electrotécnica, Inst. Sup. de Eng. do Porto,*
*R. Dr Ant. Bernadino Almeida, 4200-072 Porto, Portugal*
jtm@isep.ipp.pt

Abstract: This paper addresses the robotic manipulator synthesis problem considering multiple design criteria simultaneously. This is a complex problem suitable for the application of multi-objective genetic algorithms. Thus, an hierarchical multi-objective genetic algorithm is proposed to generate a robot structure and corresponding manipulating trajectories. The design aim is to minimize the trajectory space ripple, the initial and final torques while optimizing the mechanical structure. Simulation results are presented concerning the solution of a structure synthesis problem with the optimization of three objectives.

Keywords: Genetic Algorithms, Robotics, Trajectory Planning, Optimization, Multi-Objective.

## 1. INTRODUCTION

Genetic Algorithms (Gas) have been successfully applied to solve a wide range of engineering problems (Chambers, 2000) since the pioneering work from (Holland, 1992). This article addresses two optimizations problems: the design of a robotic manipulator structure and the trajectory planning. The latter aims to obtain a continuous motion which allows the manipulator to move from a pre-defined starting point to a desired end-point within the workspace.

Several single-objective based techniques have been proposed to solve these and related robotic problems such as: trajectory planning, manipulator structure and collision avoidance (Chocron and Bidaud, 1997), (Han *et al.*, 1997), (Kim and Khosla, 1992) and (Gallant and Boudreau, 2000). However, some of these problems are multi-objective in nature and would benefit greatly to be solved with multi-objective optimization techniques.

Multi-objective techniques using GAs have been increasing in relevance as a research area. Goldberg (Goldberg, 1989) suggested the use of a GA to solve multi-objective problems and since then other investigators have been developing new methods, such as: multi-objective genetic algorithm (MOGA) (Fonseca and Fleming, 1995), non-dominated sorted genetic algorithm (NSGA) (Deb, 2001) and niched Pareto genetic algorithm (NPGA) (Horn *et al.*, 1994), among many other variants (Coello and Carlos, 1999; Coello *et al.*, 2005). This paper proposes the use of a multi-objective method to optimize a manipulator trajectory. The proposed method is based on a GA adopting direct kinematics. The optimal structure front is the one that minimizes the objectives.

The paper outline is as follows: section 2 formulates the problem and the GA based method for its resolution. Section 3 presents several simulations results involving different robots, objectives and workspace settings. Finally, section 4 outlines the main conclusions.

## 2. PROBLEM AND ALGORITHM FORMULATION

This study considers robotic manipulators that are required to move from an initial point up to a given final position. In the experiments 1 up to 4 dof planar manipulators were adopted with rotational and prismatic joints. The arm link length are in the range $[0.1, 1]$ m with increments of $0.1$ m, and the robot rotational joints are free to rotate $2\pi$ rad. Therefore, the manipulator workspace is a circle with a $4$ m maximum radius. In what concerns the *structure* generator, it is adopted a hierarchical GA, with 3 GAs to perform the search.

The hierarchical EA is adopted in this work with four EAs, (see figure 1). A MOEA is used to evaluate the robot's structure, *structure* generator. For each structure population element three single GAs are executed. Two GAs are used to calculate the initial and final configurations of the trajectory. The third GA determines the intermediate configurations between the two points calculated previously, called *trajectory* generator, in order to find an optimal robot path. Therefore, for each structure three GAs are executed and the best fitness for each GA are used to form the three objective values of the structure solution.

### 2.1 *Representation*

The robotic structure string is represented in (1) where $J_i$ represents the type of the $i^{th}$ joint (this variable can take two values: R for rotational and P for prismatic joints) and $l_i$ is the $i^{th}$ link length, in the range $[0, 1]$ m with allowed increments of $0.1$ m. In order to limit the computational time the number of *dof* is limited to $k \leq 4$. All values used in this work are encoded through real values except the type of the robotic link.

$$S_{\{J:l\}} = \{(J_1^{(T)}, l_1^{(T)}), \ldots, (J_k^{(T)}, l_k^{(T)})\} \quad (1)$$

On the other hand, the initial and final configuration are encoded as (2).

$$\{q_1^{(T)}, \ldots, q_k^{(T)}\} \quad (2)$$

Finally, the path is encoded, directly, as strings in the joint space to be used by the GA as:

$$\{(q_1^{(1,T)}, \ldots, q_k^{(1,T)}), \ldots, (q_1^{(n-2,T)}, \ldots, q_k^{(n-2,T)})\} \quad (3)$$

In the generation $T$, the $i^{th}$ joint variable for a robot intermediate $j^{th}$ position is $q_i^{(j,T)}$, the chromosome is constituted by $n-2$ genes (configurations) and each gene is formed by $k$ values. The values of $q_i^{(j,0)}$ are initialized in the range $]-2\pi, 2\pi]$ for R-joints and $[0.1, 1]$ m for the case of P-joints. It should be noted that the initial and final configurations have not been encoded into the string because this configuration remains unchanged throughout the trajectory search. Without losing generality, for simplicity, it is adopted a normalized time of $\Delta t = 0.1$ s between two consecutive configurations, because it is always possible to perform a time re-scaling.

### 2.2 *Operators in the multi-objective genetic algorithm*

Initial populations are generated at random. The search is then carried out among these populations. The different operators used in the *trajectory* planning are reproduction, crossover and mutation, as described in the sequel. Successive generations of new strings are reproduced on the basis of their fitness function. In this case, it is used a rank selection to select the strings from the old population, up to the new population with $\sigma_{\text{share}} = 0.01$ and $\alpha = 2$. To promote population diversity a metric count is used. This metric uses all solutions in the population independently of their rank to evaluate every fitness function. For the crossover operator it is used the simulated binary crossover (*SBX*)(Deb, 2001). After crossover, the best solutions (among both parents and children) are chosen to form the next population. The mutation operator consists on several actions namely, changing the type of the joint, modifying the link length and changing the joint variable. The mutation operator replaces one gene value with a given probability using equation (4) at generation $T$, where $N(\mu, \sigma)$ is the normal distribution function with average $\mu$ and standard deviation $\sigma$.

$$q_i^{(j,T+1)} = q_i^{(j,T)} + \mathrm{N}(0, 1/\sqrt{2\pi}) \quad (4)$$

The operators used for the *structure* optimization are: duplication operator, $p_d$, that divide one link in two links with same length; the fusion operator, $p_r$, that join two links; and the mutation operator that changes the length link following equation (5). In all operators the link length restrictions are kept. At the end of each structure GA iteration, the next structure population is selected based on the maximin scheme structure (Solteiro Pires *et al.*, 2005).

$$l_i^{(T+1)} = l_i^{(T)} + \mathrm{N}(0, 1/\sqrt{2\pi}) \quad (5)$$

### 2.3 *Evolution criteria*

Three indices $\{f_{\tau_i}, f_{\tau_f}, f_q\}$ (6) are used to qualify the evolving trajectory robotic manipulators. These criteria are minimized by the planner to find the optimal Pareto front. Before evaluating any solution all the values such that $|q_i^{((j+1)\Delta t,T)} - q_i^{(j\Delta t,T)}| > \pi$ are

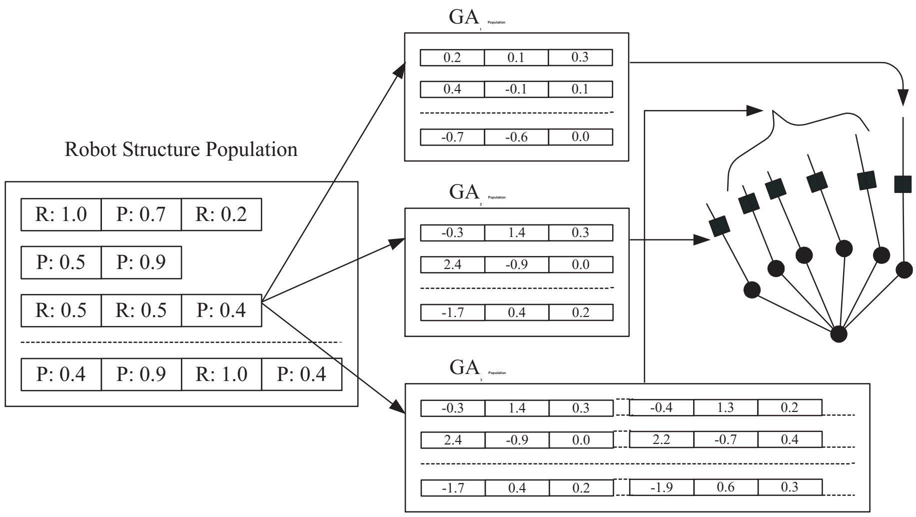



Fig. 1. Hierarchical Genetic Algorithm

readjusted, adding or removing a multiple value of $2\pi$, in the strings.

$$f_\tau = g\sum_{j=1}^{k}\sum_{i=j}^{k} m_i \sum_{p=1}^{i-1} l_p(\cos(\theta_p)(j \le p) + 0.5\cos(\theta_i)) \tag{6a}$$

$$\theta_p = \sum_{i=1}^{p} q_i \tag{6b}$$

$$f_q = \sum_{j=1}^{n}\sum_{l=1}^{k}\left(\dot{q}_l^{(j\Delta t,T)}\right)^2 \tag{6c}$$

The gravitational torque (6a) of extreme positions is used in order to minimize the energy required particularly when the manipulator has long stops points.

The joint distance $f_q$ (6c) is used to minimize the manipulator joints travelling distance. For a function $y = g(x)$ the curve length is defined by:

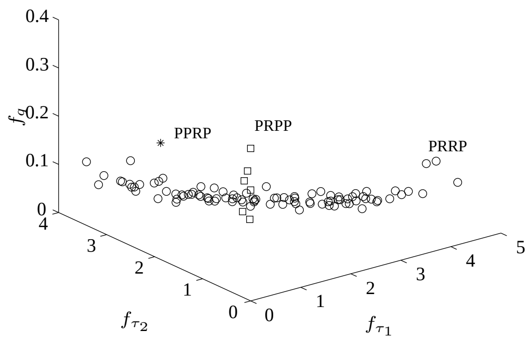
$$\int [1 + (\mathrm{d}g/\mathrm{d}x)^2]\mathrm{d}x \tag{7}$$

and, consequently, to minimize the curve length distance the following simplified expression is adopted:

$$\int (\mathrm{d}g/\mathrm{d}x)^2\mathrm{d}x = \int \dot{g}^2\mathrm{d}x. \tag{8}$$

## 3. SIMULATION RESULTS

The experiments consist on moving a robotic arm from the starting point $A \equiv \{1.0, 0.8\}$ up to the final point $B \equiv \{-0.4, 1.2\}$. The simulations results were achieved by using the following GA settings, with $n = 9$ configurations, $T^{(c,t,s)} = \{200, 15000, 1200\}$ for configuration, trajectory and structure generations, respectively. The population size is $pop_{\text{size}}^{(c,t,s)} = \{200, 100, 100\}$, duplication probability $p_d = 0.1$, fusion probability $p_r = 0.1$, crossover probability $p_c = 0.8$ and mutation probability $p_m = 0.1$.

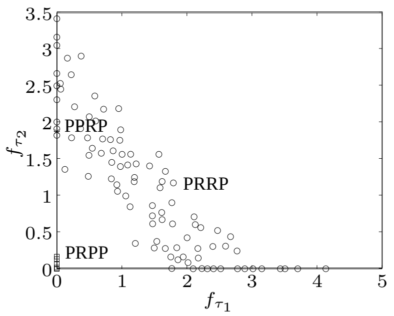



(a) Pareto optimal front

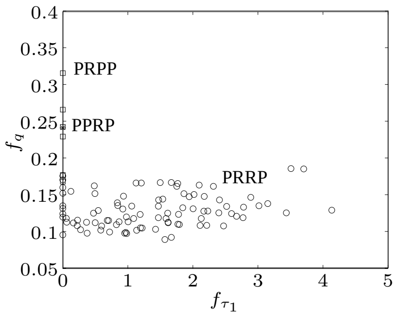
Fig. 2. Pareto optimal front $\{f_{\tau_1}, f_{\tau_2}, f_q\}$ and Pareto optimal front plane projections: $\{f_{\tau_1}, f_{\tau_2}\}$, $\{f_{\tau_1}, f_q\}$ and $\{f_{\tau_2}, f_q\}$

The algorithm determines the non-dominated front maintaining a good distribution of solutions along the Pareto front (figure 2) since the spacing index (Schott, 1995) is SP $= 0.072$ and the Minimal Distance Graph index (Solteiro Pires *et al.*, 2005) is MDG $=$ $0.122$. However, solutions along $f_q$ objective are few relatively to the others objectives because the maximin sorting scheme is used without a scale normalization in all objectives.

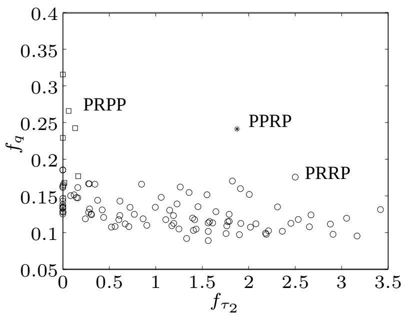
The extreme performance solutions of the front are different due to the objectives considered. Between these extreme optimal solutions several others were found, that have a intermediate behavior, and which can be selected according with the importance of each objective. The achieved front structures obtained in the simulation are depicted in table 1, in which P and R means prismatic and rotational joint, respectively. Table 2 presents for each objective, the best structur obtained. In figure 3 to 5 are shown same different structures of the front. In (a) figures are illustrated




(b) $\{f_{\tau_1}, f_{\tau_2}\}$ plane projection




(c) $\{f_{\tau_1}, f_q\}$ plane projection




(d) $\{f_{\tau_2}, f_q\}$ plane projection

Fig. 2. Pareto optimal front $\{f_{\tau_1}, f_{\tau_2}, f_q\}$ and Pareto optimal front plane projections: $\{f_{\tau_1}, f_{\tau_2}\}$, $\{f_{\tau_1}, f_q\}$ and $\{f_{\tau_2}, f_q\}$ (*cont.*)

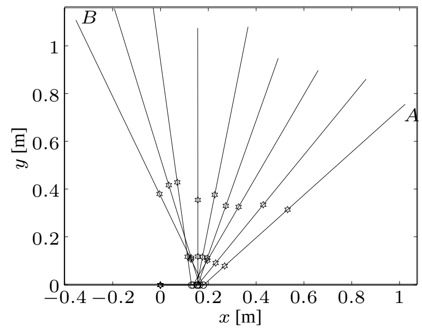
Table 1. Number of non-dominated solutions for structs

| Structure | Number of solutions |
|---|---|
| PRRP | 93 |
| PRPP | 6 |
| PPRP | 1 |

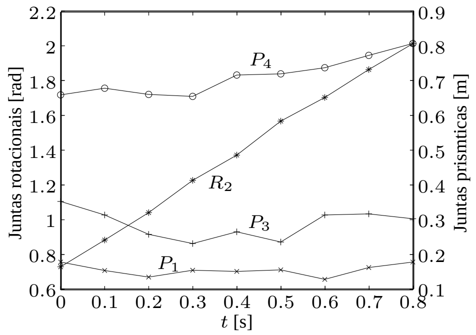
Table 2. Best Structurs for each Objective

| Objective | Structure | $l_1$ cm | $l_2$ cm | $l_3$ cm | $l_4$ cm |
|---|---|---|---|---|---|
| $f_{\tau_1}$ | PRPP | 18 | 12 | 36 | 99 |
| $f_{\tau_2}$ | PRPP | 10 | 10 | 38 | 96 |
| $f_q$ | PRRP | 11 | 14 | 39 | 97 |

the successive configurations of the structures where a circle means a rotational joint and a star represents a prismatic joint. In (b) figures it can be seen the joint position of trajectory *vs.* time where $J_i$ represents the joint type $J = \{\mathrm{R}, \mathrm{P}\}$ for the link $i = \{1, \ldots, 4\}$. The rotational and prismatic scales are in the left and right side of the graphs.

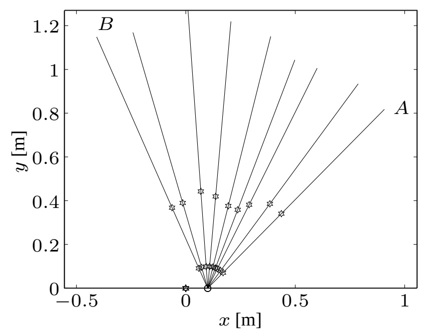



(a) Successive configurations

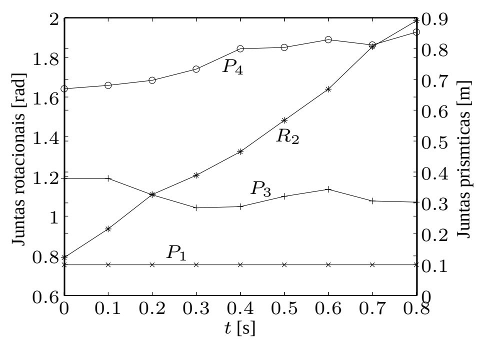



(b) Joint position of trajectory *vs.* time

Fig. 3. Best PRPP robot manipulator for $f_{\tau_2}$ Objective



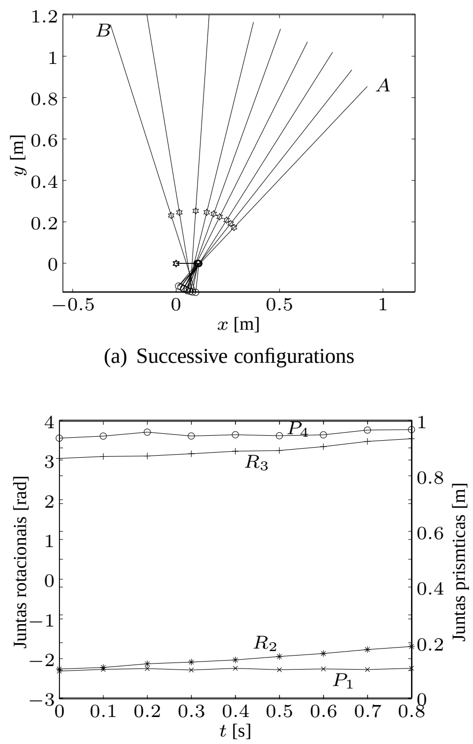

(a) Successive configurations




(b) Joint position of trajectory *vs.* time

Fig. 4. Best PRPP robot manipulator for $f_{\tau_2}$ Objective

Analyzing the final number of axis, we conclude that the larger the number of dof the better the robot ability to maneuver and to reach the desired points. The structure have a rotational joint near of the robot




(a) Successive configurations



(b) Joint position of trajectory *vs.* time

Fig. 5. Best PRRP robot manipulator for $f_{\tau_2}$ Objective base. From (b) figures can be seen that the joint cross distance are very near of the optimal.

## 4. SUMMARY AND CONCLUSIONS

A multi-objective genetic algorithm robot structure and trajectory planner, based on the kinematics approach, was proposed. The multi-objective genetic algorithm is able to reach optimal solutions regarding the optimization of multiple objectives. The algorithm is able to reach Pareto front and the solutions presents a low gravitational binary at the start and end positions and a reduced ripple in the space trajectory evolution according to objective selected. Furthermore, the algorithm determines the robot structure more adaptable to a given number and type of tasks, maintaining good manipulating performances. Simulation results were presented considering the optimization of three simultaneous objectives.

## 5. REFERENCES


Chambers, Lance D. (2000). *The Practical Handbook of Genetic Algorithms: Applications, Second Edition*. Chapman & Hall/CRC; 2 edition.

Chocron, O. and P. Bidaud (1997). Evolutionary algorithms in kinematic design of robotic system. In: *IEEE/RSJ International Conference on Intelligent Robotics and Systems*. Grenoble, France. pp. 279–286.

Coello, Carlos A. Coello, Aguirre, Arturo Hernández and Zitzler, Eckart, Eds.) (2005). *Evolutionary Multi-Criterion Optimization, Third International Conference, EMO 2005, Guanajuato, Mexico, March 9-11, 2005, Proceedings*. Vol. 3410 of *Lecture Notes in Computer Science*. Springer.

Coello, Coello and A. Carlos (1999). A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information Systems* **1**(3), 269–308.

Deb, Kalyanmoy (2001). *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley-Interscience Series in Systems and Optimization.

Fonseca, C. M. and P. J. Fleming (1995). An overview of evolutionary algorithms in multi-objective optimization. *Evolutionary Computation Journal* **3**(1), 1–16.

Gallant, Marise and Roger Boudreau (2000). The synthesis of planar parallel manipulators with prismatic joints for an optimal, singularity-free space. *Journal of Robotic Systems* **19**(1), 13–24.

Goldberg, David E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison – Wesley.

Han, Jeongheon, W. K. Chung, Y. Youm and S. H. Kim (1997). Task based design of modular robotic manipulator using efficient genetic algorithm. In: *IEEE Int. Conf. on Robotics and Automation*. Albuquerque, New Mexico. pp. 507–512.

Holland, John H. (1992). *A Adaptation in Natural and Artificial Systems: An Introduction analysis with Applications to Biology , Control, and Artificial Intelligence*. third ed.. MIT Press.

Horn, J., N. Nafploitis and David Goldberg (1994). A niched pareto genetic algorithm for multi-objective optimization. Proceedings of the First IEEE Conference on Evolutionary Computation. pp. 82–87.

Kim, Jin-Oh and Pradeep K. Khosla (1992). A multi-population genetic algorithm and its application to design of manipulators. In: *IEEE/RSJ Int. Conf. on Intelligent Robotics and Systems*. Raleight, North Caroline. pp. 279–286.

Schott, Jason R. (1995). Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization. Master's thesis. Massachusetts Institute of Technology, Department of Aeronautics and Astronautics. Cambridge, Massachusetts.

Solteiro Pires, E. J., P. B. de Moura Oliveira and J. A. Tenreiro Machado (2005). Multi-objective maximin sorting scheme. In: *Conference on Evolutionary Multi-criterion Optimization – EMO 2005*. Springer-Verlag, Lecture Notes in Computer Science Vol. 3410. Guanajuanto, México. pp. 165–175.