ANALELE UNIVERSITĂȚII
"EFTIMIE MURGU" REȘIȚA
ANUL XVII, NR. 1, 2010, ISSN 1453 - 7397

Cornel Secară, Dan Dumitriu

Direct Search Based Strategy for Obstacle Avoidance of a Redundant Manipulator

This paper presents an iterative direct search based strategy for redundancy resolution. The end-effector of the redundant manipulator achieves the imposed task of following the contour of a curve, while fulfilling two other performance criteria: obstacle avoidance and minimization of the sum of joint displacements. The objective function to minimize is the sum of joint displacements, while the obstacle avoidance and end-effector task are expressed as non-linear constraints. The proposed direct search based strategy is iterative; the joint configuration computed in the previous step represents the current point around which the generalized pattern search algorithm is successfully performed.

Keywords: *Redundant manipulator, direct search strategy, obstacle avoidance*

1. Introduction

A redundant manipulator has more joints than necessary for achieving the end-effector task. The mapping from the task coordinates to the joint coordinates is not unique, thus the inverse kinematics problem for redundant robots has an infinite number of solutions, if no additional constraints are involved. But, the "unnecessary" degrees of freedom (DOF) supposed by redundancy can be used to fulfill or optimize additional performance criteria [1], by introducing appropriate constraints, i.e., additional relations between the joint coordinates. By introducing the additional criteria, the redundancy inverse kinematics problem takes the form of a non-linear optimization problem with non-linear constraints.

The human body and the snake are two examples of redundancy provided by the nature. Since robotics is very often inspired by nature, the snake-like robotic arm, called also serpentine robot, represents a hyper-redundant robot which can be used for bridge inspection [2,3]. Humanoid robots are also composed of several redundant parts: the arms, the legs, a 3-DOF articulated spine [4], etc. As far as classical redundant manipulators for manufacturing and industry are concerned, one can distinguish parallel manipulators (e.g., planar 2-DOF redundant parallel manipulator [5]) and serial ones, such as the 4-DOF planar SCARA type manipulator considered as case study in this paper [6,7].

Two main categories can be distinguished regarding the redundancy resolution:

• Methods based on the direct geometric model, i.e., the direct relation (1) between the end-effector configuration vector $\mathbf{x} = [\ x_1\ x_2\ ...\ x_m\ ]^{\mathrm{T}}$ and the joint coordinates vector (angles vector $\boldsymbol{\theta} = [\ \theta_1\ \theta_2\ ...\ \theta_n\ ]^{\mathrm{T}}$, for rotation joint case):

$$\mathbf{x} = f(\boldsymbol{\theta}), \tag{1}$$

where n is the number of DOF and m is the workspace dimension, with $n > m$ in the case of redundancy. These redundancy resolution methods based on the direct geometric model consist of minimizing a cost function expressed, e.g., as the sum of joint displacements between two/all successive end-effector locations, while imposing the end-effector task and other additional constraints (obstacle avoidance) as a non-linear constraint vector function to be fulfilled. This cost function minimization under non-linear constraints can be solved using either genetic algorithms [6-12], direct search [1,13], neural networks [12], or fuzzy techniques [14].

• Methods based on inverse kinematics, i.e., the inverse of the direct geometric model (1):

$$\boldsymbol{\theta} = f^{-1}(\mathbf{x}), \text{ or } \delta\boldsymbol{\theta} = \mathbf{J}^{-1}\delta\mathbf{x}, \tag{2}$$

where $\mathbf{J}(\boldsymbol{\theta}) = \dfrac{\delta f(\boldsymbol{\theta})}{\delta\boldsymbol{\theta}}$ is the manipulator's Jacobian $m\times n$ matrix, resulting from the direct kinematics equation deduced from (1): $\delta\mathbf{x} = \mathbf{J}(\boldsymbol{\theta})\delta\boldsymbol{\theta}$. Unfortunately, in the case of redundancy, the Jacobian $\mathbf{J}$ is not a square matrix since $n > m$. In this situation, instead of $\mathbf{J}^{-1}$, a pseudoinverse of Jacobian $\mathbf{J}^{+} = \mathbf{J}^{\mathrm{T}}(\mathbf{J}\mathbf{J}^{\mathrm{T}})^{-1}$ is built, where $\mathbf{J}^{+}$ is indeed a $n\times n$ square matrix. However, the simple use of this pseudoinverse $\mathbf{J}^{+}$ is not sufficient since this right-hand side term represents just the least norm solution, which guarantees only the end-effector task accomplishment and some minimization of the sum of joint displacements, but not the additional obstacle avoidance constraint. To deal with the obstacle avoidance additional constraint as well, two basic methods can be distinguished [15]: the Gradient Projection Method (GPM) and Extended Jacobian Method (EJM). GPM obtains the solution of the non-linear optimization problem with non-linear constraints, by adding to the least norm solution $\mathbf{J}^{+}\delta\mathbf{x}$ (which ensures that the end-effector follows the task path), the so-called null space solution, which takes into account additional constraints and consists of a self-motion of the links in the joint space only, without any effect on the end-effector task configuration vector. With respect to the EJM, this second method defines $n-m$ additional constraints including the obstacle avoidance for the given task, thus the relationship between the joint space and the end-effector space becomes non-redundant, and the extended Jacobian is now a $n \times n$ square matrix which can be easily inverted.

Recent approaches have tried to solve the redundancy problem by combining a method based on inverse kinematics with one based on the direct geometric model. Such combined methods can avoid the joint angle drift problem caused by the disadvantage that the pseudoinverse control is not repeatable. Da Graça Marcos et al. [16,17] propose a technique that combines the closed-loop pseudoinverse method with genetic algorithms (GAs). An open-loop GA based on the direct geometric model is used to find the initial joint configuration, and also to compare the results provided by the new combined method.

2. Iterative (sequential) versus global search minimization strategy

This paper presents an iterative direct geometric model method, based on the direct search technique only. Let us explain the reason why an iterative strategy is considered to be good enough compared to a global strategy. Section 3 will explain why the direct search is used herein instead of more sophisticated techniques, such as GAs or neural networks.

The goal of the redundancy problem is to accomplish the imposed end-effector task of following the contour of a curve, while ensuring obstacle avoidance and minimizing the sum of joint displacements required for accomplishing the end-effector task. This non-linear optimization problem with non-linear constraints can be formulated as follows:

- Minimize the cost function f expressed as the sum of the joint displacements required for accomplishing the end-effector task

$$f = \sum_{i=1}^{N_\mathrm{p}-1} \| \boldsymbol{\theta}_{i+1} - \boldsymbol{\theta}_i \| = \sum_{i=1}^{N_\mathrm{p}-1} f_i \tag{3}$$

while

- imposing the end-effector task

$$\| \mathbf{x}_i - \mathbf{x}_{\mathrm{d},i} \| \le \varepsilon_\mathrm{d}, \quad i = 1 \div N_\mathrm{p}, \tag{4}$$

- ensuring the obstacle avoidance

$$\min(d_{kl})_i \ge d_0, \quad i = 1 \div N_\mathrm{p}, \quad k = 1 \div N_\mathrm{CCP}, \quad l = 1 \div N_\mathrm{O}. \tag{5}$$

In relations (3)-(5), N_p is the number of imposed end-effector postures on the curve contour to be followed, $\boldsymbol{\theta}_i$ is the joint coordinates vector corresponding to the real end-effector i-th posture $\mathbf{x}_i$, while $\mathbf{x}_{\mathrm{d},i}$ is the desired end-effector i-th posture on the curve contour to be followed, with ε_d the admissible error between real $\mathbf{x}_i$ and desired $\mathbf{x}_{d,i}$ end-effector postures. With respect to the obstacle avoidance constraint (5), the minimum of the d_{kl} distances must be greater than a desired d_0 distance imposed by the user. The d_{kl} distances are calculated between the k-th Configuration Control Point (CCP) and the l-th obstacle, where N_{CCP} is the number of the CCP and N_O is the number of obstacles. The CCP are imposed by the user on the manipulator structure.

The redundancy problem of minimizing the cost function (3), subject to constraints (4) and (5), represents a global non-linear optimization problem. It is global in the sense that the minimization process concerns f, i.e., the sum of the joint displacements for all N_p imposed end-effector postures. Most methods based on the direct geometric model proposed in the literature [6,8-11,13] concern the global optimization problem (3)-(5).

However, this paper concerns an iterative method based on the direct geometric model, leading to a sequential minimization of the cost function f, more precisely at each time step $i = 1 \div N_{p-1}$ one has to minimize the piece f_i of the overall cost function f, where f_i is the norm of joint displacements vector necessary to move the end-effector of the redundant manipulator from the posture $\mathbf{x}_i$ to the next posture $\mathbf{x}_{i+1}$:

$$f_i = \left\| \boldsymbol{\theta}_{i+1} - \boldsymbol{\theta}_i \right\|, \qquad (6)$$

while imposing the end-effector task (4) and ensuring the obstacle avoidance (5) at time t_i .

The solution obtained after a sequence of N_{p-1} iterative direct search minimizations of the f_i defined by (6) will be considered satisfactory since also the GPM and EJM, methods based on inverse kinematics, perform a sequential optimization, being considered as valuable in the literature [15]. Moreover, a previous paper [7] provided better results when using a sequence of N_{p-1} iterative GA-based minimizations of f_i , instead of using GPM inverse kinematics method.

Without proving it rigorously, let us explain why a sequential optimization can provide a result quite close to the result of the global optimization, using the principle of optimality introduced by Richard Bellman as axiomatic basis for the dynamic programming [18]. According to Bellman's principle of optimality, an optimal "strategy" can only be constituted of optimal actions, implying that, for every initial state and initial action, the next actions must represent optimal actions in relation to the intermediary state that results from the first action [18,19]. By "strategy" one understands here a certain succession of actions from the initial to the final state. This axiomatic principle of optimality of Bellman appears almost as a truism. In fact, if an optimal strategy includes a non-optimal action, it seems possible to replace this non-optimal action with the optimal one, thus apparently improving the strategy. However, this replacement could affect the performance of neighboring actions, and finally the new strategy could turn out to be worse. Nevertheless, numerous numerical tests have proved the practical efficiency of Bellman's principle of optimality.

Figure 1 illustrates the iterative direct search based strategy proposed in this paper. The sequential optimization consists of optimizing each action in strict relation with the previous one and with the perspective of the actions to be taken from now on, i.e., minimizing f_i starting from $\boldsymbol{\theta}_i$ issued from the previous minimization of f_{i-1} .

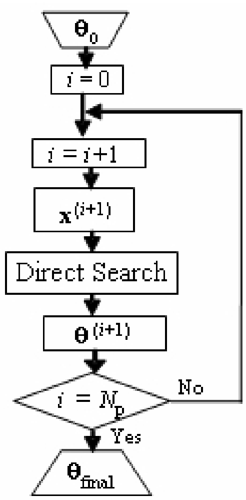

Figure 1. Iterative scheme of the direct search strategy

2.2. Direct Search versus GA

Let us decide now what search method to use to successively minimize fi given by (6) at each time t_i . A previous paper [7] used GAs to minimize f_i , but GAs has two main disadvantages: 1) they do not necessarily find the exact global optimum, convergence is not guaranteed even to local minima; 2) convergence may be slow, requiring a large, unknown number of evaluations of the cost function, thus GAs cannot assure constant optimization response times [20-22].

These two disadvantages of GAs can be eliminated by simply using the direct search method. This basic method searches a set of points around the current point, looking for one where the value of the cost function is lower than the value at the current point. Of course, as for GAs, it does not require any information about the gradient of the cost function.

Clearly, the use of the direct search method is appropriate only if the dimension of the problem is not too big, so that to avoid the exponential increase of the computational time. For manipulator tasks which are not so fast, e.g., following the contour of a circle in a few minutes, one can speak about obtaining the redundancy problem solutions in almost real time.

3. Case study: 4-dof planar redundant manipulator

The simulations were performed on a laboratory model of planar redundant manipulator, possessing four DOF (see Figure 2). This experimental model was realized at the Laboratory of Robotics-Mechatronics Group of Institute of Solid Mechanics of Romanian Academy [6].

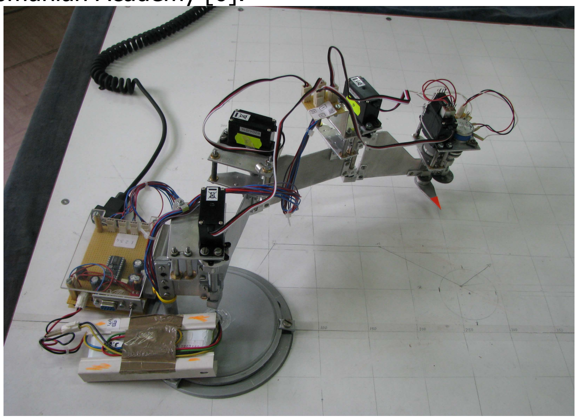

Figure 2. Laboratory model [6]

The proposed aim is to generate the references (positions and orientations) of the end-effector along the contour of a circle of radius r, whose surface is considered to be restrictive for all four elements of the manipulator structure. The operational space dimension is, in this case, $m=3$ because the position and orientation of the end-effector are both taken into account. Thus, the degree of redundancy is $n-m=1$.

The initial posture of the manipulator is illustrated in Figure 3 and is given by the following measures:

$$\boldsymbol{\theta}_0 = [0.72 \;\; 5.49 \;\; 5.55 \;\; 3.93];$$
$$l_1 = 0.12; \;\; l_2 = 0.12; \;\; l_3 = 0.10; \;\; l_4 = 0.05; \;\; x_0 = 0; \;\; y_0 = 0; \qquad (7)$$
$$r = 0.03; \;\; x_c = 0; \;\; y_c = 0.2.$$

where θ_0 is the initial joint coordinates vector expressed in radians, l_1, l_2, l_3 and l_4 are the lengths of the four links (in meters), x_0 and y_0 are the manipulator base coordinates, r is the radius of the restriction circle and x_c and y_c are its centre coordinates.

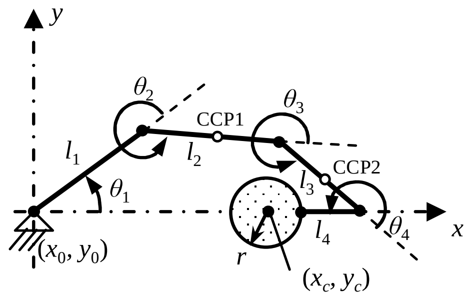

Figure 3. Initial manipulator configuration

The end-effector references generation is a function of the sampling step of generation, i :

$$x_d^{(i)} = x_c + r\cdot\cos(i\cdot\Delta\alpha);$$
$$y_d^{(i)} = y_c + r\cdot\sin(i\cdot\Delta\alpha); \qquad (8)$$
$$\Sigma\theta_d^{(i)} = 5\pi + i\cdot\Delta\alpha;$$

where $\Delta\alpha$ is the angular step of generation.

The end-effector coordinates, obtained using the direct geometric model, have the following expressions:

$$x^{(i)} = \sum_{j=1}^{4} l_j \cdot \cos\left(\sum_{k=1}^{j} \theta_k^{(i)}\right);$$
$$y^{(i)} = \sum_{j=1}^{4} l_j \cdot \sin\left(\sum_{k=1}^{j} \theta_k^{(i)}\right); \qquad (9)$$
$$\Sigma\theta^{(i)} = \sum_{j=1}^{4} \theta_j^{(i)}.$$

The CCP (N_{CCP} =2) are placed in the middle of second element and, respectively, in the middle of third element of the manipulator, and the desired distance imposed by the user is $d_0 = 0.015$ [m].

The imposed positioning and orientation error of the end-effector is $\varepsilon_d = (0.001\ \ 0.001\ \ 0.1^o)$.

The angular step of generation is $\Delta\alpha=3^\circ$ and, thus, the number of strategy steps to entirely cover the circle is $N_p=120$. The vector that produces the lower and upper bounds of the direct search variables is constant for every i-th step: $\Delta\boldsymbol{\theta}=[4^\circ\ 7^\circ\ 8^\circ\ 4^\circ]$.

Figure 4 shows the results for the redundant manipulator following the imposed circle. The simulation was performed using the Direct Search Toolbox of MATLAB 7.1, with the following parameters [23]:

- poll method: pattern search algorithm; polling order: consecutive
- initial mesh size = 1; maximum mesh size: infinite
- mesh contractor factor = 0.5
- mesh expansion factor = 2
- mesh contraction factor = 0.5

The simulation results show a better performance compared to the results of using redundancy resolution methods with linearized solutions. For instance, the sum of joint angles displacements for all sampling steps in our case is 13.44 radians, smaller than 15.06 obtained using the Gradient Projection Method with an artificial repulsive field working in the null-space of the Jacobian matrix [15].

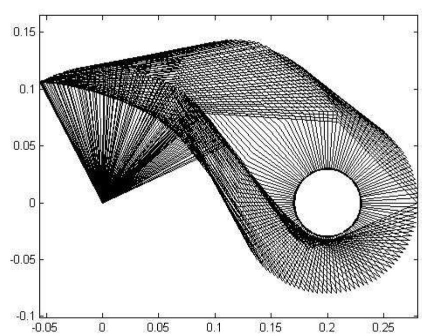

Figure 4. Simulation results

5. Concluding remarks

The sequential optimization approach proposed in this paper consists of using the direct search method so that to minimize, at each time step, the norm fi of the joint displacements vector. For the case study of a 4-DOF planar redundant manipulator, the sequential optimization proved to be almost as good as a global optimization minimizing the sum of all joint displacements vectors. For problems with this order of dimension, direct search proved to be a simple choice, avoiding the convergence issues associated, for example, with GAs.

ACKNOWLEDGEMENTS: The authors gratefully acknowledge the financial support of the National Authority for Scientific Research (ANCS, UEFISCSU), Romania, through PN-II research project no. 106/2007.

References

[1] Kapoor, C., Cetin, M., Tesar, D., 1998, *Performance Based Redundancy Resolution with Multiple Criteria*, Proceedings of DETC98, ASME Design Engineering Technical Conference, Atlanta, Georgia (September 1998).

[2] Sofge, D., Chiang, G., 2001, *Design, Implementation, and Cooperative Coevolution of an Autonomous/Teleoperated Control System for a Serpentine Robotic Manipulator*, Proceedings of the American Nuclear Society Ninth Topical Meeting on Robotics and Remote Systems, Seattle, Washington (March 2001).

[3] Choset, H., 2000, *Bridge Inspection with Serpentine Robots*, Final Report for Highway IDEA Project 56, Carnegie Mellon University.

[4] Roos, L., Guenter, F., Guignard, A., Billard, A., 2006, *Design of a Biomimetic Spine for the Humanoid Robot Robota*, IEEE / RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, Pisa, Italy (February 2006).

[5] Yiu, Y.K., Li, Z.X., 2001, *Dynamics of a Planar 2-dof Redundant Parallel Robot*, International Conference on Mechatronics Technology, Singapore (June 2001)

[6] Secară, C., Chiroiu, V., Dumitriu, D., 2009, *Obstacle Avoidance by a Laboratory Model of Redundant Manipulator using a Genetic Algorithm based Strategy*, Proceedings of IV-th National Conference The Academic Days of the Academy of Technical Sciences in Romania, Iassy, Romania (November 2009), pp. 199-204.

[7] Secară, C., Vlădăreanu, L., 2010, *Iterative strategies for obstacle avoidance of a redundant manipulator*, WSEAS Transactions on Mathematics, Vol. 9 (3), pp. 211-221.

[8] Mitsi, S., Bouzakis, K.-D., Sagris, D., Mansour, G., 2008, *Determination of Optimum Robot Base Location considering Discrete End-effector Positions by means of Hybrid Genetic Algorithm*, Robotics and Computer-Integrated Manufacturing, Vol. 24 (1), pp. 50-59.

[9] Mitsi, S., Bouzakis, K.-D., Sagris, D., Mansour, G., 2005, *A Multi-objetive Optimum Robot Base Placement using a Hybrid Genetic Algorithm*, The Ninth IFToMM International Symposium on Theory of Machines and Mechanisms, Bucharest, Romania (September 2005).

[10] Nearchou, A., 1998, *Solving the Inverse Kinematics Problem of Redundant Robots Operating in Complex Environments via a Modified Genetic Algorithm*, Mech. Mach. Theory, Vol. 33 (3), pp. 273-292.

[11] Tian, L., Collins, C., 2003, *Motion Planning for Redundant Manipulators using a Floating Point Genetic Algorithm*, Journal of Intelligent and Robotic Systems, Vol. 38, pp. 297-312.
[12] Du, X., Chen, H.-H., Gu, W.-K., 2005, *Neural Network and Genetic Algorithm Based Global Path Planning in a Static Environment*, Journal of Zhejiang University Science, Vol. 6A (6), pp. 549-554.
[13] 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 (3), pp. 239-244.
[14] Ham, C., Qu, Z., Johnson, R., 2000, *Robust Fuzzy Control for Robot Manipulators*, IEEE Proc.-Control Theory Applications, Vol. 147 (2), pp. 212-216.
[15] Secară, C., 2008, *Control Strategies for Obstacle Avoidance by Redundant Manipulators*, Proceedings of the Romanian Academy, Series A: Mathematics, Physics, Technical Sciences, Information Science, Vol. 9 (1), pp. 55-60.
[16] Da Graça Marcos, M., Tenreiro Machado, J.A., Azevedo-Perdicoúlis, T.-P., 2009, *Trajectory planning of redundant manipulators using genetic algorithms*, Communications in Nonlinear Science and Numerical Simulation, Vol. 14 (7), pp. 2858-2869.
[17] Da Graça Marcos, M., Tenreiro Machado, J.A., Azevedo-Perdicoúlis, T.-P., 2010, *An evolutionary approach for the motion planning of redundant and hyper-redundant manipulators*, Nonlinear Dynamics, Vol. 60 (1-2), pp. 115-129.
[18] Bellman, R., 1957, *Dynamic Programming*, Princeton University Press, Princeton, NJ.
[19] Chevalier, A., 1977, La Programmation Dynamique, Dunod, Paris.
[20] Goldberg, D.E., 1989, *Genetic Algorithms, Search, Optimization & Machine Learning,* Addison Wesley, Reading, MA.
[21] Deb, K., 2001, *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons Ltd.
[22] Coley, D., 1999, *An Introduction to genetic algorithms for scientists and engineers*, World Scientific Press.
[23] MATLAB 7.1 (Release 14 with Service Pack 3), *Genetic Algorithm and Direct Search Toolbox*, The MathWorks.

Addresses:

- Dr.Eng. Cornel Secară, Institute of Solid Mechanics, Romanian Academy – 15 C-tin Mille, Bucharest 10141, Romania, corsecus@yahoo.com
- Dr.Eng. Dan Dumitriu, Institute of Solid Mechanics, Romanian Academy – 15 C-tin Mille, Bucharest 10141, Romania, dumitriu@imsar.bu.edu.ro