Computing (2007)
DOI 10.1007/s00607-006-0190-2
Printed in The Netherlands

Computing

Two-dimensional motion-planning for nonholonomic robots using the bump-surfaces concept

E. K. Xidias, Patras, **P. N. Azariadis**, Syros, and **N. A. Aspragathos**, Patras

Received December 12, 2005; revised April 16, 2006
Published online: March 7, 2007

Abstract

In this paper, a new method is introduced for finding a near-optimal path of a nonholonomic robot moving in a 2D environment cluttered with static obstacles. The method is based on the Bump-Surfaces concept and is able to deal with robots represented by a translating and rotating rigid body. The proposed approach is applied to car-like robots.

AMS Subject Classifications: 65D17, 68020, 93C10, 70B10, 70E17.

Keywords: Path planning, motion design, robotics, nonholonomic robots, bump-surfaces.

1. Introduction

The problem of motion planning for mobile robots among stationary obstacles usually deals with holonomic vehicles which can move in any direction of the free configuration space [1], which represents all the possible robot configurations. Using the configuration-space concept the moving object (robot) is transformed to a point and the obstacles are expanded appropriately. Recently, nonholonomic mobile robots have received an increased attention by the researchers working in this field. This interest is motivated by the large class of mechanical systems that belong in this category and the challenging problem of finding an optimal path among obstacles [2].

In most of the cases, a nonholonomic robot has features similar to that of a car, i.e., a rectangular-shaped body and a limited steering angle, complicating the motion-planning problem due to (a) the shape of the robot and (b) the existence of nonholonomic constraints which impose a lower-bounded turning radius. This means that the path should be conformal to strict curvature requirements that general path planners cannot satisfy [3]. The proposed methods for solving the motion planning problem of nonholonomic robots can be classified in two main categories: search-based methods and potential-field based methods.

The search-based methods [4]–[7] generally construct a roadmap in two phases. In the preprocessing phase a roadmap is developed where no restriction is placed on the nonholonomic robot. In the second phase, the edges of the roadmap are reduced by taking into account the nonholonomic constraints. These methods are global in nature, meaning that they search over the entire configuration space [7]. However, the resulted path is polygonal (e.g., a sequence of line segments) which is an unpleasant feature because the robot is forced to make a complete stop each time it meets a curvature discontinuity at the vertices of this "polygonal path".

Potential-fields methods [8]–[11] force the robot to move according to the influence of an artificial potential field produced by the goal configuration and the obstacles. The main limitation arises from the existence of local minima in the resulted field, where no descent direction exists for the robot to follow.

Recently, the last two authors of this paper introduced the concept of *Bump-surfaces* in order to resolve the problem of motion-planning in two dimensions (2D) [12]. In that work the robot is considered as a point moving in an environment with arbitrary static obstacles which have been expanded appropriately. Therefore the produced solution does not take into account any further degree of freedom that corresponds i.e., to rigid body rotations. The search method is global and it is based on an energy-minimizing criterion applied to a variational curve design problem in a subspace $[0, 1]^3 \subset \Re^3$ embedded in $\Re^3$. The final path is represented by a C^2 curve satisfying optimally the given motion (or energy) criteria.

A similar in-spirit approach is proposed by Hofer and Pottman in [13]. They follow also a conservative approach considering the robot as a moving ball onto the six-dimensional manifold of the Euclidean Group SE(3) of rigid body motions, and they use a variational curve design algorithm for minimizing a quadratic energy function. The resulting solution does not utilize all the available degrees of freedom of the moving robot, while the proposed approach requires the definition of intermediate configurations in the workspace.

This paper extends the original work presented in [12] by employing all the available configurations to resolve the problem of motion-planning of nonholonomic robots moving in an environment with arbitrary obstacles. In the present work the robot has a 2D shape with finite dimensions (i.e., it is not considered as a point anymore), while the shape and size of the obstacles remain unchanged (contrarily to the standard configuration-space approaches). The new motion-planning method is focused on planning forward motion only, i.e., without backing-up manoeuvres, and takes into account generic nonholonomic constrains. Similarly to [12] the entire robot environment and all possible robot configurations are represented through a single Bump-Surface while the final path is determined by solving an energy-minimizing motion-design problem.

2. Problem statement

Let a nonholonomic robot with two rear and two directional front wheels, which is modelled as a polygon moving in an a priori known 2D environment called *workspace*. The robot workspace is cluttered with static obstacles which have arbitrary size, shape and location. The motion-planning problem addressed in this paper con-

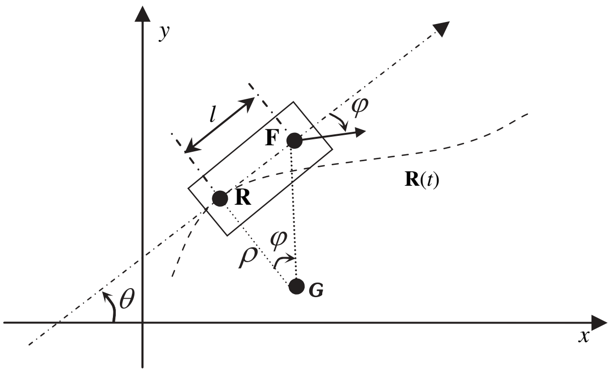

Fig. 1. A car-like robot

cerns the determination of a path connecting an initial and a final configuration in this workspace.

The configuration of a nonholonomic robot in the workspace is given by a triple $(u, v, \theta) \in \Re^2 \times [0, 2\pi)$, where (u, v) are the coordinates of the rear axle midpoint **R** with respect to a fixed frame, and θ is the orientation of the robot, i.e., the angle between the x-axis and the main axis of the robot. The *steering angle* $0 \leq |\varphi| \leq \varphi_{max}$, with $|\varphi| = arctan(\frac{l}{\rho})$, is defined by the main axis of the robot and the velocity vector of the midpoint **F** of the two front wheels. Finally ρ is the radius of curvature at point **R** (see Fig. 1) [1], [15].

The unknown path is determined by taking into account the following criteria and constraints:

(i) The path should not intersect with the obstacles.
(ii) The length of the path should be minimal.
(iii) The path should have a lower-bounded turning radius.
(iv) The path should be smooth without sharp corners and with optimal parameterization.

The robot's path is represented by a B-Spline curve in order to take into advantage the local attributes and the well-known numerical stability of the respective computational implementations [14].

3. Formulation of the motion planning objectives

The motion design problem described earlier is formulated in this section as a global energy-minimizing problem with nonlinear constrains.

3.1. General considerations

The nonholonomic robot $\boldsymbol{A}$ is modelled as a convex polygon moving in a two-dimensional environment $\boldsymbol{W} = [0, 1]^2 \in \Re^2$. Typically the corresponding configuration space should be represented by a cylindrical three-dimensional manifold $\Re^2 x\ [0,\ 2\pi)$ [1]. However we take advantage of the fact that the robot has a maximum steering angle φ_{max} which can be directly connected to the path's radius of curvature ρ [1]. Therefore all feasible configurations of $\boldsymbol{A}$ are represented onto the Bump-surface manifold S through a family of one-parametric curves lying on S.

In the present method we consider that the midpoint of the rear wheels of the non-holonomic robot traces a path $R = \mathbf{R}(t) = (u(t), v(t))$ given as a B-Spline curve

$$\mathbf{R}(t) = \sum_{i=0}^{K-1} N_{i,d}(t)\mathbf{p}_i, \quad 0 \le t \le 1 \tag{1}$$

defined in the parametric space of S (without loss of generality we consider that the parametric space of S is the actual workspace $\boldsymbol{W}$). Here, K is the number of control points $\mathbf{p}_i$, $N_{i,d}$ is the B-Spline basis function and d is the curve degree. Our variational curve design problem is focused in the definition of the $(K-2)$ control points $\mathbf{p}_i$ such that curve R satisfies the motion-planning conditions **2(i)–2(iv)**. The first and last control points, namely $\mathbf{p}_0$ and $\mathbf{p}_{K-1}$, are fixed to the initial and final position of the midpoint of the robot rear wheels. The orientation of the nonholonomic robot in each point of the path has the same direction with the tangent vector of R at that point. If fixed robot orientations are required in the two path end-points then we consider these two orientations equal to vectors $\mathbf{T}_0$ and $\mathbf{T}_{K-1}$, respectively, where $\|\mathbf{T}_0\| = \|\mathbf{T}_{K-1}\| = 1$.

3.2. Satisfying the objectives 2(i) and 2(ii)

Following the results from [12] a valid path that avoids the obstacles should be searched in the "flat" areas of the Bump-surface. A path that "climbs" the bumps of the Bump-surface results to an invalid path in $\boldsymbol{W}$ that penetrates the obstacles. By construction, the arc length of R approximates the length of its image $S(\mathbf{R}(t))$ on S, as long as R does not penetrate the obstacles. Therefore it is reasonable to search for a "flat" path on S in order to satisfy both objectives **2(i)** and **2(ii)**.

The arc length of the image of curve R onto S is given by [16]:

$$L = \int_{o}^{1} \sqrt{E\left(\frac{du}{dt}\right)^2 + 2F\frac{du}{dt}\frac{dv}{dt} + G\left(\frac{dv}{dt}\right)^2}\, dt, \tag{2}$$

where E, F and G are the quantities of the first fundamental form of S.

In order to take into account the shape of the nonholonomic robot $\boldsymbol{A}$ which should not intersect with the obstacles, we measure the "flatness" of the vertices $\mathbf{a}_j$ of $\boldsymbol{A}$ onto S. Thus, similarly with midpoint $\mathbf{R}(t)$, each vertex $\mathbf{a}_j$ follows a curve $\mathbf{a}_j = \mathbf{a}_j(t)$ on $\boldsymbol{W}$ where its image on S is given by $S(\mathbf{a}_j(t))$. We measure the "flatness" of $S(\mathbf{a}_j(t))$ through

$$H_j = \int_0^1 S_z(\mathbf{a}_j(t))dt, \quad j = 1, \ldots, n_A \ \text{ with } \ n_A \ge 2, \tag{3}$$

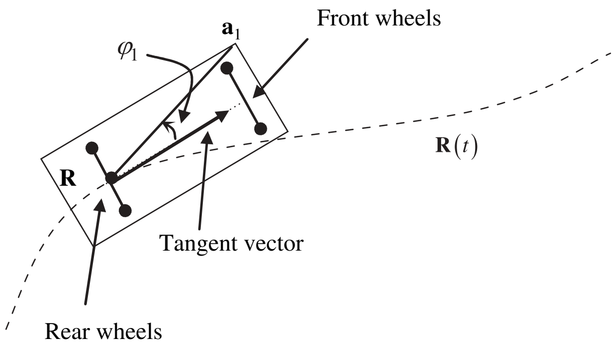

Fig. 2. The robot's position and orientation with respect to a given path R

where, $S_z(\mathbf{a}_j(t))$ denotes the z-coordinate of the image of $\mathbf{a}_j$, $j = 1, \ldots, n_A$ on S, and n_A is the number of vertices that approximate the shape of $\boldsymbol{A}$. The distance between every vertex $\mathbf{a}_j$, $j = 1, \ldots, n_A$ of $\boldsymbol{A}$ and the corresponding midpoint $\mathbf{R}(t)$ of the rear axle is constant. The same holds also for the angle between vectors $\left(\mathbf{R}(t) - \mathbf{a}_j(t)\right)$ and $\mathbf{R}'(t)$ (tangent vector of R) as shown in Fig. 2 for $j = 1$. Therefore the location of every vertex $\mathbf{a}_j$ is determined with respect to $\mathbf{R}(t)$.

Putting together Eqs. (2) and (3), the minimizer of the *external energy* of R

$$E_{ext.} = \sum_{j=1}^{n_A} H_j + L \tag{4}$$

with respect to $\mathbf{p}_i$ ($i = 1, \ldots, K-1$) leads to a path which optimally satisfies the conditions **2(i)** and **2(ii)**.

3.3. Satisfying the objective 2(iii)

The third objective imposes a minimum turning radius ρ_{min} in the path. Hence, the midpoint $\mathbf{R}(t)$ of the rear robot-axle should follow a path whose curvature $k(t)$ is smaller or equal than $\frac{1}{\rho_{min}}$. This requirement is expressed by

$$k(t) \leq \frac{1}{\rho_{min}} = k_{max} \quad \text{or} \quad \frac{\left\|\mathbf{R}'(t) \times \mathbf{R}''(t)\right\|}{\|\mathbf{R}'(t)\|^3} \leq k_{\text{max}}. \tag{5}$$

Since, the robot's path is a planar curve $R = \mathbf{R}(t) = (u(t), v(t)) = \sum_{i=0}^{K-1} N_{i,d}(t)(x_i, y_i)$, the roots of the derivative of $k(t)$ are computed by solving

$$k'(t) = 0 \quad \text{or} \quad \left(\frac{u'(t)v''(t) - u''(t)v'(t)}{\left(u'(t)^2 + v'(t)^2\right)^{3/2}}\right)' = 0. \tag{6}$$

In this way we determine the parameters $t_i,\ i = 1, \ldots, n$ which correspond to the path points of maximum curvature. Hence, objective **2(iii)** can be expressed by

$$\max\,(k(t_1), \ldots, k(t_n)) \leq k_{max}. \tag{7}$$

Note: The roots of Eq. (6) are determined using the Regula Falsi method [17] between the inflection points [14] of the B-spline curve C.

3.4. Satisfying the objective 2(iv)

Objective **2(iv)** expresses the requirement for a smooth path C without sharp corners and loops and with an optimal parameterization, allowing, thus, for a smooth robot-motion with constant velocity. These requirements are expressed by

$$E_{int} = \int_0^1 \left\| \boldsymbol{R}'(t) \right\|^2 dt + \int_0^1 \left\| \boldsymbol{R}''(t) \right\|^2 dt. \tag{8}$$

The minimizer of E_{int} with respect to $\mathbf{p}_i$ $(1 \leq i \leq K-2)$ subject to the constraint Eq. (7) corresponds to a path satisfying optimally both objectives **2(iii)** and **2(iv)**.

Note: When an initial and final robot orientation is given through $\mathbf{T}_0$ and $\mathbf{T}_{K-1}$ (see Sect. 3.1) then we introduce into Eq. (8) two more terms, i.e.,

$$E_{int} = \int_0^1 \left\| \mathbf{R}'(t) \right\|^2 dt + \int_0^1 \left\| \mathbf{R}''(\mathbf{t}) \right\|^2 dt + \left\| \mathbf{T}_0 - \frac{\mathbf{R}'(0)}{\|\mathbf{R}'(0)\|} \right\|^2 + \left\| \mathbf{T}_{K-1} - \frac{\mathbf{R}'(1)}{\|\mathbf{R}'(1)\|} \right\|^2. \tag{9}$$

3.5. Forming the overall motion-planning problem

Taking the above into account we conclude that the minimizer of the following combined energy E_{comb}:

$$E_{comp} = a_1 E_{ext} + a_2 E_{int}, \tag{10}$$

$$\text{Subject to } \max\,(k(t_1), \ldots, k(t_n)) \leq k_{max}$$

with respect to $\mathbf{p}_i$ $(1 \leq i \leq K-2)$, is a continuous collision-free path which satisfies optimally the motion-planning objectives **2(i)–2(iv)**. The scalars $a_1 + a_2 = 1$, $0 \leq a_1, a_2 \leq 1$ are weight factors. For example, higher values for a_1 enforce the optimization method to search for shorter paths ignoring to some degree their smoothness and vice versa.

4. An energy-minimizing algorithm for motion design with Bump-surfaces

Equation (10) represents a nonlinear objective function with a non-linear constraint. In order to effectively resolve this problem in a computer-based environment the integrals in Eq. (2), (3) and (8) are discretized and approximated through finite sums. Then an evolutionary optimization method is employed to solve the obtained optimization problem.

4.1. Discrete representation of the minimization problem

The robot's path R is approximated by $N_p - 1$ sequential chords. The r-th point of R is given by

$$\mathbf{R}(t_r) = \sum_{i=0}^{K-1} N_{i,d}(r\Delta t)\mathbf{p}_i = \sum_{i=0}^{K-1} N_{i,d}(r\Delta t)(x_i, y_i), r = 0, \ldots, N_p - 1, \tag{11}$$

where $\Delta t = \frac{1}{N_p - 1}$. Equation (2) is replaced by the sum of the squared lengths of the images of the chords of R on S. Hence,

$$L^* = \sum_{r=0}^{N_p-2} \left\| \mathbf{S}\left(\mathbf{R}\left(t_{r+1}\right)\right) - \mathbf{S}(\mathbf{R}(t_r)) \right\|^2. \tag{12}$$

In a similar fashion, Eq. (3) is replaced by

$$H^* = \sum_{j=1}^{n_A} H_j = \sum_{j=1}^{n_A} \sum_{r=0}^{N_{p-1}} S_z\left(u_r^j, v_r^j\right), \quad n_A \geq 2, \tag{13}$$

where, $\left(u_r^j, v_r^j\right)$ is the r-th point of each $\mathbf{a}_j(t)$, $j = 1, \ldots, n_A$. Thus, Eq. (4) now becomes

$$E_{ext}^* = \sum_{r=0}^{N_p-2} \left\| \mathbf{S}\left(\mathbf{R}\left(t_{r+1}\right)\right) - \mathbf{S}(\mathbf{R}(t_r)) \right\|^2 + \sum_{j=1}^{n_A} \sum_{r=0}^{N_p-1} S_z\left(u_r^j, v_r^j\right), \quad n_A \geq 2. \tag{14}$$

Similarly, Eq. (8) is replaced by

$$E_{int}^* = \sum_{r=0}^{N_p-1} \left\| \mathbf{R}'(t_r) \right\|^2 + \sum_{r=0}^{N_p-1} \left\| \mathbf{R}''(t_r) \right\|^2 \tag{15}$$

and the final optimization problem is written as

$$E_{comb}^* = a_1 E_{ext}^* + a_2 E_{int}^*, \tag{16}$$

$$\text{Subject to } \max\left(k(t_1), \ldots, k(t_n)\right) < k_{max}.$$

Let us note here, that the first and second derivative vectors in Eq. (15) are computed analytically with respect to Eq. (1) for $t = t_r$.

4.2. The optimization algorithm

In this paper, we employ Genetic Algorithms (GAs) [18] in order to derive a reliable and accurate solution of the optimization problem expressed by Eq. (16). The proposed method utilises a uniform crossover, a Gaussian mutation operator and a Roulette Wheel selection. In addition, the following fitness function is used in order to minimize E^*_{comb}:

$$E_{ga} = \frac{1}{E^*_{comb} + Q + X}, \tag{17}$$

where, Q is given by

$$Q = \frac{L^*}{D} \mathrm{e}^{-g \frac{D}{L^*}}. \tag{18}$$

L^* is the arc length of the image of curve R onto the S (Eq. (12)), g is the maximum allowed number of control points and D is the Euclidean length of the straight line connecting the start and the destination point. Equations (18) are incorporated in the fitness function in order to allow the algorithm to determine the optimal number of control points that define the robot's path. Finally, the curvature constraint is given by

$$X = \frac{\max(k(t_1), \ldots, k(t_n))}{k_{max}}. \tag{19}$$

Each chromosome represents a possible path for the robot as a sequence of control points which define the B-spline curve of Eq. (1). The coordinates (x_i, y_i) of each control point are the *genes* of the chromosome. The initial population consists of a number of chromosomes having genes with coordinates randomly selected within $[0, 1]^2$.

5. Implementation and results

The proposed method has been implemented on a Pentium IV 3.2 GHz PC using Matlab. The following examples concern the motion design of a car-like robot with dimensions equal to 0.075×0.025 environment units (the distance between its two axes is 0.07 units). The grid size is set to 75×75, the maximum number of control points is set to $g = 12$ and the general GA's control parameters are defined as follows: *population size* = 100, *number of generations* = 90, *crossover rate* = 0.7 and *probability of boundary mutation* = 0.0075. Finally, $N_p = 100$ and $a_1 = 0.7$, $a_2 = 0.3$ in order to increase the importance of the first two motion-planning objectives. Finally, a 2-degree B-spline curve and a (2, 2)-degree B-spline surface are used to represent the robot path and the B-surface, respectively.

Scene I: Figure 3a shows a car-like robot which has steering-angle limit 45°. The 2D environment is cluttered with 11 obstacles of arbitrary size and shape. The control polygon of the final path consists of 5 control points. The processing time is 163 seconds.

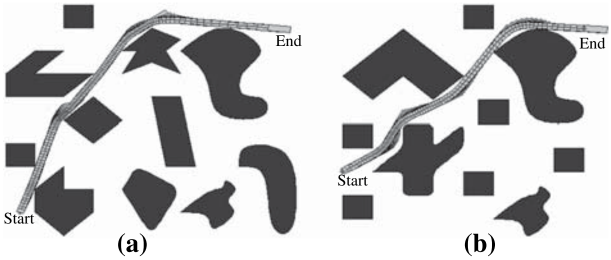

Fig. 3. The solution path $\mathbf{R}(t)$ (red curve) between a start and end configuration

Scene II: Figure 3b shows a car-like robot which has steering-angle limit 45°. The 2D environment is cluttered with 10 obstacles of arbitrary size and shape. The control polygon of the final path consists of 6 control points. The processing time is 149 seconds.

Both examples concern motion design problems in complicated environments where the robot is requested to travel a long distance avoiding static obstacles with arbitrary size, shape and location. The computed paths $\mathbf{R}(t)$ are smooth and satisfy all the motion-planning criteria **2(i)–2(iv)**. Although, using GAs it is not possible to guarantee that a valid solution path will be obtained after each run, all our experiments have shown that typically one is able to get a near-optimum solution within the first or the second run (for an extensive discussion on this issue the reader is referred to [12, Sect. 3.6.3]).

6. Conclusions

In this paper, a new method is presented for motion-planning of nonholonomic robots. The proposed method employs all the available degrees of freedom in the motion design process. Computer simulations have shown that the new method works efficiently in complicated environments with arbitrarily shaped obstacles.

Acknowledgments

This work is financed by the Research Committee of the University of Patras as a part of the research project "An Optimal Motion Planning for a Robot based on Computational Geometry" under the K. Karatheodoris frame. University of Patras is a partner in the EU-funded FP6 Innovative Production Machines and Systems (I*PROMS) Network of Excellence.

References

[1] Latombe, J. C.: Robot motion planning. Boston: Kluwer Academic Publishers 1991.
[2] Kolmanovsky, I., McClamroch, N. H.: Developments in nonholonomic control problems. IEEE Control Systems, pp. 20–36 (1995).
[3] Jiang, K., Seneviratne, L. D., Earles, S. W. E.: A shortest based path planning algorithm for nonholonomic mobile robots. J. Intell. Robotics Sys. *24*, 347–366 (1999).

[4] Vougioukas, S. G.: Optimization of robots paths computed by randomized planners. In: Proc. IEEE Int. Conf. on Robot Automat., Barcelona, Spain, pp. 2160–2165 (2005).
[5] Pruski, A., Rohmer, S.: Robust path planning for non-holonomic robots. J. Intell. Robotic Sys. *18*, 329–350 (1997).
[6] Song, G., Amato, N. M.: Randomized motion planning for car-like robots with C-PRM. In: Proc. 2001 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), pp. 37–42 (2001).
[7] Divelbiss, A. W., Wen, J. T.: A path space approach to nonholonomic motion planning in the presence of obstacles. IEEE Trans. Robot. Automat. *13*(3), 443–451 (1997).
[8] Barraquand, J., Langlois, B., Latombe, J. C.: Numerical potential field techniques for robot path planning. In: Proc. IEEE 5th Int. Conf. on Advanced Robot., Pisa, Italy, pp 1012–1027 (1991).
[9] Volpe, R., Kholsa, P.: Artificial potential with elliptic isopotential contours for obstacle avoidance, 28th IEEE Conf. on Decision and Control, Los Angeles, CA, pp. 180–185 (1987).
[10] Rimon, E., Koditschek, E.: Exact robot navigation using artificial potential functions. IEEE Trans. Robot. Automat. *8*, 501–518 (1992).
[11] Bemporad, A., Luca, A., Oriolo, G.: Local incremental planning for a car-like robot navigating among obstacles. IEEE Int. Conf. on Robotics and Automation, Minneapolis, USA, pp. 1205–1211 (1996).
[12] Azariadis, P., Aspragathos, N.: Obstacle representation by bump-surfaces for optimal motion-planning. J. Robotics Autonomous Sys. *51*(2–3), 129–150 (2005).
[13] Hofer, M., Pottmann, H.: Energy-minimizing splines in manifolds. Trans. Graphics *23*(3), 284–293 (2004).
[14] Piegl, L., Tiller, W.: The NURBS book. Berlin Heidelberg: Springer 1997.
[15] LaValle, S. M.: Planning algorithms. University of Illinois 1999–2004.
[16] do Carmo, M. P.: Differential geometry of curves and surfaces. Prentice-Hall 1976.
[17] Cooper, L., Steinberg, D.: Introduction to methods of optimization. W. B. Saunders 1970.
[18] Goldberg, D. E.: Genetic algorithms in search, optimization and machine learning. Addison Wesley 1989.

E. K. Xidias and N. A. Aspragathos
Department of Mechanical
and Aeronautics Engineering
University of Patras
Patras, 26500 Greece
e-mails: {xidias, asprag}@mech.upatras.gr
http://www.mead.upatras.gr/en/default.htm

P. N. Azariadis
Department of Product
and Systems Design Engineering
University of the Aegean
Ermoupolis, Syros, 84100 Greece
e-mail: azar@aegean.gr
www.syros.aegean.gr/en/default.htm