DEVELOPED HYBRID GENETIC ALGORITHM FOR OPTIMIZING REVERSE ENGINEERING METHODS

Gabriel Mansour, Sevasti Mitsi, Konstantinos-Dionysios Bouzakis, Dimitrios Sagris & Emmanouel Varitis

Laboratory for Machine Tools and Manufacturing Engineering, Mechanical Engineering Department, Aristoteles University of Thessaloniki, 54124, Greece

Corresponding author: Gabriel Mansour, mansour@eng.auth.gr

Abstract: An important area in Reverse Engineering applications is the combination of multiple scans of a 3D product to model the object. Moreover the registration refinement of multiple range images is a crucial step in multi-view 3D modeling. In the present paper a hybrid optimization method is developed to align point clouds, without any user-applied initial alignment. The proposed method combines a genetic algorithm with a quasi-Newton algorithm and furthermore a constraints handling method is involved. Several free-form point clouds are used to verify the accuracy and the reliability of the proposed method and two characteristic examples are presented. The applied examples point clouds are retrieved using a Coordinate Measuring Machine (CMM), which is not restrictive for the developed optimizing methodology.
Key words: Optimization, reverse engineering, genetic algorithm, point clouds, CMM.

1. INTRODUCTION

Nowadays Reverse Engineering (RE) is a popular method of object modeling. RE is performed through a scanning procedure, which aims to obtain a point cloud. The most common difficulty in this process is to register into one part, various views of the same object, which have been obtained by a 3D scanning device (Coordinate Measurement Machine-CMM, laser scan, light sectioning).
The most common applications where the registration method takes place are the modeling of an object through the combination of multiple point clouds and in the inspection process of a CAD model [Pottmann et al., 2002]. Since laser scanners have a limited field of view, in order to obtain a complete representation and modeling of an object it is necessary to collect data from several different locations that must be transformed into a common reference coordinate system. The point clouds are necessary to have a common area. In the inspection process of a CAD model the object is measured with a measuring device, such as CMM, and the acquired point cloud is compared with the CAD model of the same object. The CAD model, which represents the original object, is located in a different position with respect to the reference coordinate system than the 3D data point. For the success of shape inspection, the problem is to find the transformation that aligns the point cloud to the CAD model. This method gives the ability to provide 3D CAD-to-part verification with complete and accurate analysis.
The automatic and semi-automatic registration procedures used in currently available commercial software are not robust and often lead to incoherent results. Many registration algorithms have been developed in recent years. The concept of the registration of two point clouds is the following. For two point clouds as input data that are usually called the model cloud and the data cloud, in a common coordinate system, the target of the registration algorithm is to find a homogeneous transformation matrix that optimally merges these two point clouds. Bels and McKay proposed the Iterative Closest Point (ICP) algorithm [Besl & McKay, 1992], where the rigid transformation between two scans is iteratively refined directly using corresponding points extracted after each refinement step. Several improvements have been proposed to this approach [Zhang, 1994] [Trucco et al., 1999], but all the ICP algorithms require an initial estimation of the rough transformation between the two point clouds and easily fall into local minima. Furthermore, the rate of ICP algorithms convergence depends on the choice of the corresponding point-pairs and the primitive location of the point clouds. Chen and Medioni [Chen & Medioni, 1992] proposed a point-to-surface algorithm, which is more accurate than the ICP algorithm, but also requires a computational expensive task [Park & Subbarao, 2003]. Some genetic algorithms (GAs) have been proposed for the registration of the point clouds. For example Chow [Chow et al., 2004] developed a dynamic GA where a new dynamic genetic operator was proposed. Galantucci et al [Galantucci et al., 2004] proposed a method based on GAs, using several spheres to subdivide the problem into regions for analysis, when

Artificial Neural Network is used to recognize the most feasible regions for alignment.

Based on the findings of previous researchers the most commonly used methods of genetic algorithms, gradient based methods and constraints handling methods are further discussed in the following sections.

1.1 Genetic algorithms

The GA is a well-known efficient global optimization algorithm, introduced by Holland [Holland, 1975] in 1975, that utilizes the concept of biological structure to natural selection and survival of the fittest. Due to the fact that the method requires no previous experience on the problem, it is applied on various problems whereof some characteristic are mentioned in [Nearchou, 1998], [Coley, 1999] and [Chapelle & Bidaud, 2004]. An initial randomized population that consists of a group of chromosomes and represent the problem variables, produces new populations through successive iterations, using various genetic operators. The common operators are selection, elitism, cross over and mutation. A function called fitness function determines when a new chromosome will replace a previous one or not, according to its worth. Through several repetitions the evolution of the individuals leads to the domination of the stronger ones. The applied operators in each step are:

Selection: The optimum chromosomes according to their fitness value are selected to contribute their content for the next generation. The common roulette-wheel selection process is used, where the selection probability of each individual is a fitness value ratio.

Elitism: In order to preserve the optimum individual of each generation for the next generation, the elitism operator is activated. The result is to keep the optimum individual of all previous generations in the current population and avoid the possibility of losing good individuals.

Crossover: Contrary to the selection operator that does not generate any new individuals but preserves the fitter ones, the crossover operator produces new chromosomes using a combination of two parent individuals. The most common method uses a single point crossover operator.

Mutation: The uniform mutation operator is applied in order to obtain variation of individuals in the evolutionary process. It operates on each binary bit of each chromosome and reverses the value of 1 to 0 and conversely.

1.2 Gradient based search methods

The gradient-based search methods [Visual Numerics, 1997] and [Kuester & Mize, 1973] use an initial variables vector guess in order to start a search of the optimum variables direction. One of these methods uses simple bounds on the variables and determines the optimum search direction through the estimation of the gradient with a finite-difference method. In each step of these algorithms the current variables vector is redirected according to the gradient, in order to achieve after several iterations a local minimum. The obtained value is a local minimum near the initial guess area, but not necessarily the global one.

1.3 Constraints handling methods

The constraints handling methods define new bounds of the variables in order to make a search algorithm faster or more focused in an area of solutions. The new intervals are used to start a new round of optimization, most commonly based on a genetic algorithm, in order to improve the final result. Generalized forms of constraints handling method as used in [Carlson, 1995] and [Michalewicz, 1995] give the ability to reduce, extend or reset the variables bounds, in order to make more efficient the search procedure.

In the present paper a hybrid optimization method [Sagris et al., 2004] and [Sagris et al., 2004] is developed to solve the automatic point clouds registration problem. The developed algorithm combines a genetic algorithm with a hill climbing method (Quasi-Newton algorithm) and furthermore a constraints handling method is involved. The usage of random numbers in genetic algorithms to produce the individuals of each generation, gives the ability to explore the whole space around the model cloud to find the best alignment position of the data cloud. With the aid of a simple genetic algorithm an initial approximate solution is found and with the Quasi-Newton optimization method this solution is guided to an optimum one. During the procedure, using a method of constraints handling, the variables limits are reduced to accelerate the optimization procedure. Although the objective function is non-differentiable the successive application of the genetic algorithm and of the gradient-based algorithm avoids the degradation of the performance of the proposed method in searching for the global minimum. The proposed algorithm works successfully for common boundary areas of point clouds. The reliability, the accuracy and the speed of the proposed method are tested through several numerical applications of various objects. Two of these applications are presented in this paper.

2. REGISTRATION OF POINT CLOUDS DATA

The point clouds registration approach is described for a couple of point clouds. For more point clouds, each step’s registration result can be combined with one new cloud. Lets set $M=[m_1, m_2, \ldots, m_n]$ and $D=[d_1, d_2, \ldots, d_k]$ the two point clouds in IR^3, called model cloud and data cloud respectively, expressed in a common fixed Cartesian coordinate system. The data cloud can be moved while the model cloud is kept fixed. The goal of the registration algorithm is to find a homogeneous transformation matrix T that best aligns the data cloud D with the model cloud M. The transformation matrix T is applied to data points in order to calculate the distance between points of the data and model clouds. For every point $T(d_j)$ (j=1,…k) of data cloud is calculated the minimum distance until all the points of the model cloud mi (i=1,…,n). Therefore, the point clouds registration can be formulated as an optimization problem, where the objective function takes into account the minimum distances of respective points of data cloud and model cloud, with respect to the transformation matrix T.

The objective function, which is minimized, can be described by:

$$F = \sum_{i=1}^{k} \left| m_i^* - T(d_i) \right|, \; k < n \tag{1}$$

where m_i^* is the model point closest of $T(d_i)$ data point.

The transformation matrix T used in equation (1), contains six parameters: the translation on x,y,z-axis and the rotation angles Roll-(α), Pitch-(β), Yaw-(γ) about x,y and z-axis respectively [Emiris, 1999].

The variables vector is,

$$\vec{V} = [x \quad y \quad z \quad \alpha \quad \beta \quad \gamma]^t = = [v_1 \quad v_2 \quad v_3 \quad v_4 \quad v_5 \quad v_6]^t \tag{2}$$

with variables bounds

$$v_{i\,\min} < v_i < v_{i\,\max}, \quad i = 1,...,6 \tag{3}$$

The x, y and z bounds are related to the data and model geometry and the orientations angles (α, β, γ) bounds are (0, 2π).

From the minimization of the objective function, the values of these unknown parameters occur.

The objective function (1) is non-differentiable and application of a gradient-based optimization algorithm degrades the performances in searching for the global minimum. In order to avoid these problems, in this paper a new algorithm is proposed, which is presented in the next section.

3. PROPOSED ALGORITHM

The described mathematical problem is solved with a hybrid method that combines a genetic algorithm (GA), a quasi-Newton algorithm (QNA) and a constraints handling method (CHM). In order to reduce the computational time, a uniformed points reduction of the point clouds is established. Moreover the center of mass of data cloud is transferred to the center of mass of model cloud to be able to reduce the bounds of each positional variable (xyz).

The basic steps of the proposed algorithm are illustrated in Figure 1. The input data for the algorithm are the two point clouds coordinates, the variables bounds and the algorithm parameters. In these parameters are included the initial parameters of the GA such as the population size, the crossover rate, the mutation rate, etc. and the number of the GA, QNA, and CHM loops. Using the equation (1) the fitness function is defined, which is used in all the steps of the algorithm.

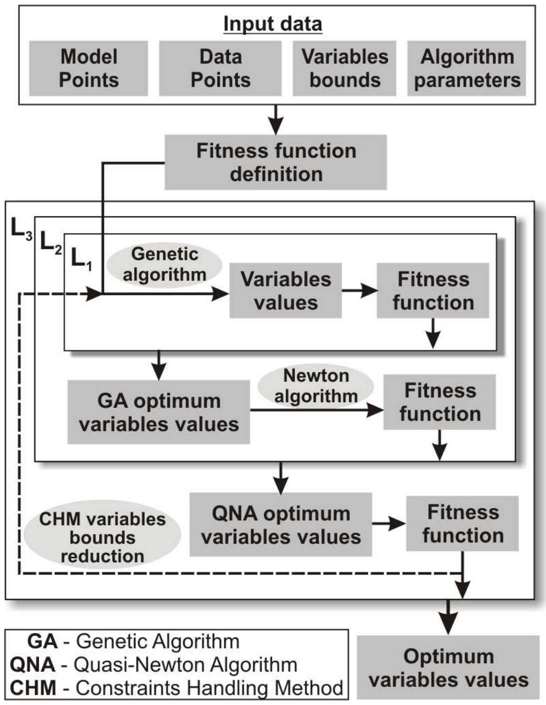

Fig. 1. Developed algorithm flowchart diagram

In the first loop of the genetic algorithm, starting populations are randomly generated to set variables values, which are used to calculate the fitness function value. Genetic algorithm uses selection, elitism, crossover and mutation procedures to create new generations. The new generations converges towards a minimum that is not necessarily the global one. After some repetitions when the maximum generations' number is achieved, the variables values corresponding to the minimum fitness function value are selected as the optimum variables values of the genetic algorithm.

The optimum GA variables values are inserted in the QNA as an initial variables vector guess. The quasi-Newton algorithm modifies the values of this vector using a finite-difference gradient method until a maximum iterations number or a local minimum is reached. Through this 'hill climbing' method a new fitness function value is obtained. The loop of QNA is applied several predefined times, including the repetition of GA loop, in order to locate several local minimums using the GA and approach the global one

using the QNA loop. When the maximum loops number of QNA is achieved, the variables values corresponding to the minimum fitness function value are selected as the optimum QNA variables values.
Afterwards, using a Constraints Handling Method, these output QNA optimum variables values are used to reduce the bounds of the variables described in equation (3). The procedure of the variables bounds reduction is illustrated in Figure 2.

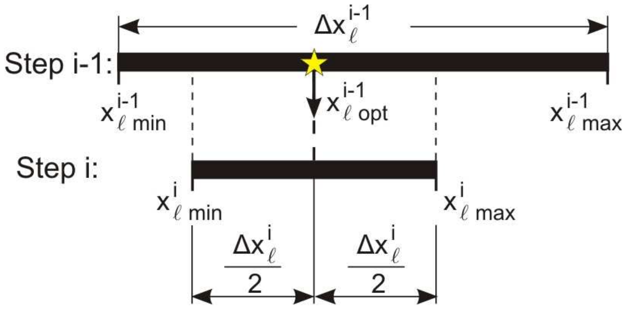

Fig. 2. Variable bounds reduction, using CHM

The optimum QNA variable value of the step i-1, $x_{\ell\,opt}^{i-1}$ ($\ell = 1,...m$), is considered as the middle of the new reduced bounds interval for the step i. The new bounds $x_{\ell\min}^{i}$ and $x_{\ell\max}^{i}$ ($\ell = 1,...m$) are given by the equations:

$$x_{\ell\min}^{i} = x_{\ell\,opt}^{i-1} - \Delta x_\ell^{i} / 2 \tag{4}$$

$$x_{\ell\max}^{i} = x_{\ell\,opt}^{i-1} + \Delta x_\ell^{i} / 2 \tag{5}$$

where the range Δx_ℓ^{i} is:

$$\Delta x_\ell^{i} = c \cdot \Delta x_\ell^{i-1} \tag{6}$$

with
$\Delta x_\ell^{i-1} = x_{\ell\max}^{i-1} - x_{\ell\min}^{i-1}$, $x_{\ell\max}^{i-1}$, $x_{\ell\min}^{i-1}$ are the variables limits of the step i-1 and c is a coefficient defined by the user. The most commonly used values of the coefficient c are 0.15 up to 0.75.
Each new bounds reduction leads to a new round of GA generations and QNA loops (see Figure 1). CHM is applied in order to reduce the variables bounds and accelerate the whole process conducting the search in a narrower area. The optimization procedure is finished when the maximum CHM loops number or the fitness function goal is achieved.
The optimum variables define the optimum homogeneous transformation matrix and the optimal location of the data point cloud. Although the fitness function is non-differentiable, the successive applications of the genetic algorithm and of the gradient-based algorithm lead to a better value of the fitness function, to a decrease of the generations number and the computational time than the ones with genetic algorithm only.

4. APPLICATIONS OF THE PROPOSED ALGORITHM

The proposed algorithm is applied in two categories of point clouds. The first category investigates the case of point clouds with same boundaries and different holes in their surfaces (Figure 3). This figure illustrates a turbines wing, which was scanned using a Coordinate Measurement Machine. The second category (see Figure 4) investigates the case of registration of a CAD model and a point cloud of the same object. The statue is called Eusy and the scanning device is a CMM. The initial number of points of the model and data clouds, the percentage of reduction and the final number of points that is used in these applications are presented in Table 1.

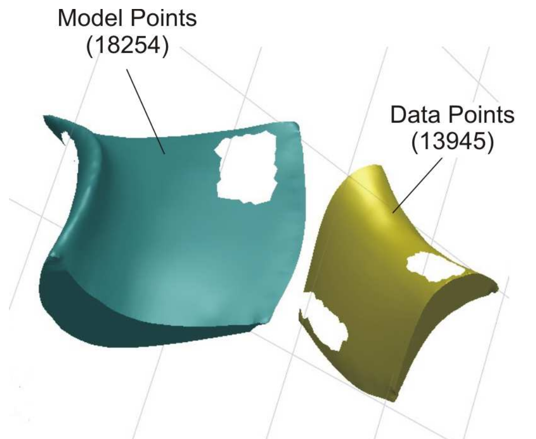

Fig. 3. Initial position of point clouds with the same boundaries and different holes

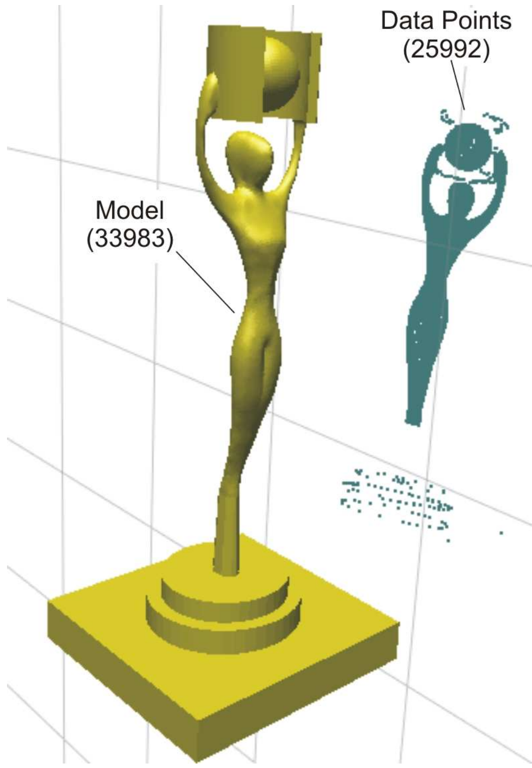

Fig. 4. Initial position of a CAD model and a point cloud

The parameters involved in all tests, mainly in GA procedure, are the same and selected as optimums through many applied tests: population of individuals=50, cross probability=70%, mutation probability=7% and the reduction of variables range in CHM=50%. All the other algorithm parameters such as the number of GAs, the number of QNAs and the number of CHMs are presented in Table 2. The loops number of GA, QNA and CHM are selected in a way that the solutions of both cases are accurate and quick enough simultaneously.

Table 1. Points number before and after the reduction

Exam-ples	Initial Points		Reduction %		Finally	
	Model	Data	Model	Data	Model	Data
Wing	18 254	13 945	90.2	97.9	1 797	293
Eusy	33 983	25 992	94.5	98.7	1 866	340

Table 2. Parameters of two numerical examples and the corresponding results of the hybrid method

Exa-mple	Parameters			Hybrid method	
	Number of loops			Compu-tational time	Fitness value
	GAs (S1)	QNAs (S2)	CHMs (S3)		
Wing	1	20	1	0:03:06	3.21E-03
Eusy	1	8	1	0:01:16	6.77E-04

The proposed algorithm is compared with a classic GA to verify the capacity and the reliability of the hybrid method. In Figure 5 is illustrated the fitness function value versus the generation number in the hybrid and a classic GA method.

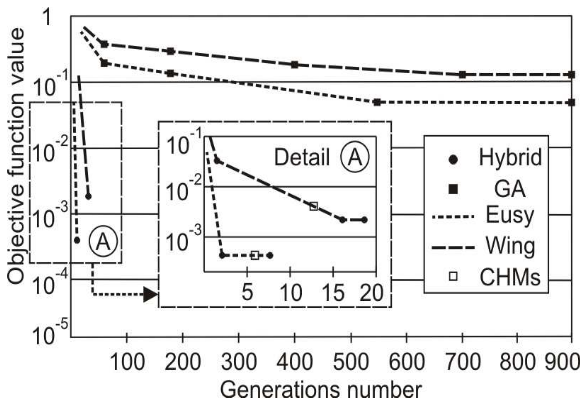

Fig. 5. Minimum objective function value versus generation's number of both examples

In both examples the fitness function value of hybrid algorithm converges in better solution in sorter time. For the hybrid method, in the example of the wing, the final alignment result was obtained after 3 generations. Additionally, in the Eusy statue the final alignment result was obtained after 16 generations. For the classic GA, in the example of the Eusy and wing, the fitness function converges in higher value after 550 and 700 generations respectively. The fitness function value, the generation number and the computational time are presented in Table 2 and Table 3. The proposed hybrid method needs less computational time than the classic GA algorithm.

Table 3. Parameters of two numerical examples and the corresponding results of the genetic method

Exa-mple	Parameters			GA method	
	Number of loops			Compu-tational time	Fitness value
	GAs (S1)	QNAs (S2)	CHMs (S3)		
Wing	900	-	-	0:15:16	1.26E-01
Eusy	900	-	-	0:06:23	5.60E-02

The registration process is also tested with commercial RE software. The automatic algorithm that is provided in this software is based on ICP algorithm and it failed to align the point clouds for all the above examples, when the point clouds were on their initial positions. When an initial manually applied approach is used, this ICP algorithm obtains an acceptable solution. Furthermore an acceptable solution can be obtained with the three points method, which is also a non-automatic procedure against the proposed one.

The registration that has been achieved with the proposed algorithm for the first category of same boundary objects and different holes is shown in Figure 6, where the positional deviation is also displayed. The maximum distance in the case of the wing is 1.104mm and the average distance 0.02mm. In the category of a point cloud and a CAD model, as illustrated in Figure 7, the maximum distance is 0.240mm and the average is 0.005mm.

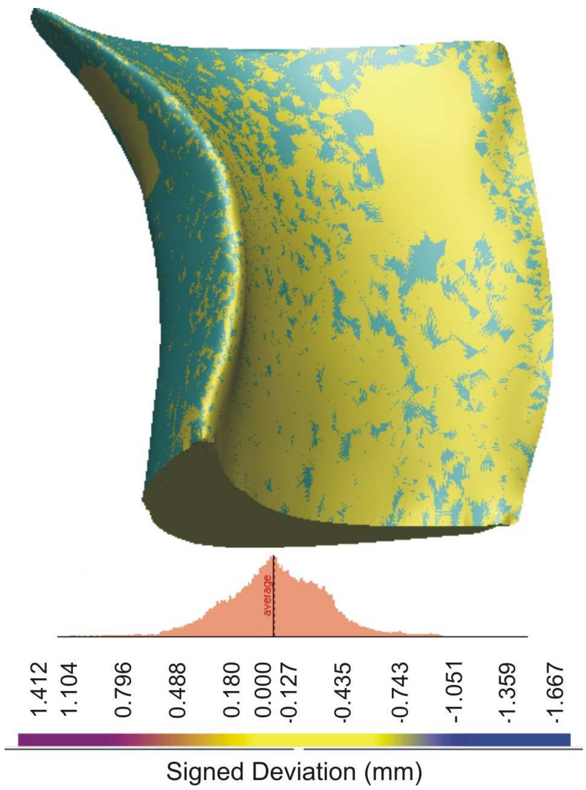

Fig. 6. Two point clouds deviation after the automatic registration process of the wing

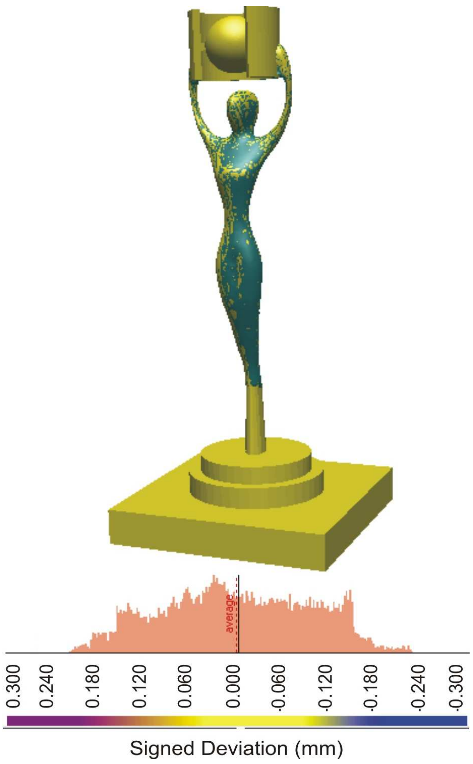

Fig. 7. Two point clouds deviation after the automatic registration process of EUSY statue

5. CONCLUSIONS

In the present paper a hybrid algorithm for registration of point clouds is developed. The method combines a genetic algorithm with a hill climbing method (quasi-Newton algorithm) and furthermore a constraints handling method is involved.

Experimental results show that the automatic proposed approach behaves much better, with higher accuracy, in comparison to common algorithms that require users' interaction. The alignment capability and the computational time are not affected by the initial positions of model and data point clouds. The algorithm leads to high accuracy results for each possible distance between the two point clouds, with a stable behavior.

6. REFERENCES

1. Besl, P.J., McKay, N.D. (1992). *A method for registration of 3D shapes*, IEEE Trans. Pattern Anal. and Mach. Intell, Vol.14, pp.329-256, ISSN:0162-8828.

2. Carlson, S. (1995). *A General Method for Handling Constraints in Genetic Algorithms*. Proceedings of the Second Annual Joint Conference on Information Science, pp.663-667.

3. Chapelle, F., Bidaud, P. (2004). *Closed form solution for inverse kinematics approximation of general 6R manipulators*. Mechanism and Machine Theory, Vol.39(3), pp.323-338, ISSN: 0094-114X.

4. Chen, Y., Medioni, G. (1992). *Object modeling by registration of multiple range images*, Image and Vision Computing, Vol.10, pp.145-155, ISSN: 0262-8856.

5. Chow, C.K., Tsui, H.T., Lee, T. (2004). Surface registration using a dynamic genetic algorithm, Pattern Recognition, Vol.37, pp.105-117, ISSN: 0031-3203.

6. Coley, D. (1999). *An Introduction to Genetic Algorithms for Scientists and Engineers*, World Scientific Press, ISBN: 981-02-3602-6.

7. Emiris, D. (1999). *Robotics*, Anosis Press, Athens, pp.138-161 (Greek), ISBN: 960-86339-1-5.

8. Galantucci, L.M., Percoco, G., Spina, R. (2004). *An artificial intelligence approach to the registration of free-form shapes*, CIRP Annals – Manufacturing Technology, Vol.53(1), pp.139-142, ISSN: 0007-8506.

9. Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, Univ. of Michigan Press, Ann Arbor, Mich., ISBN: 0-262-58111-6.

10. Kuester, J., Mize, J. (1973). *Optimization Techniques with Fortran*, McGraw-Hill Inc., ISBN: 978-0070356061.

11. Michalewicz, Z. (1995). *Genetic Algorithms, Numerical Optimization, and Constraints*, Proc. 6th International Conference on Genetic Algorithms, Pittsburgh, July 15-19.

12. Nearchou, A. (1998). *Solving the inverse kinematics problem of redundant robots operating in complex environments via a modified genetic algorithm*, Mechanism and Machine Theory, Vol.33(3), pp.273-292, ISSN: 0094-114X.

13. Park, S.-Y., Subbarao, M. (2003). *An accurate and fast point-to-plane registration technique*, Pattern Recognition Letters, Vol.24(16), pp.2967-2976, ISSN: 0167-8655.

14. Pottmann, H., Leopoldseder, S., Hofer, M., (2002). *Simultaneous registration of multiple views of a 3D object*, Int. Archives of Photogrammetry, Remote Sensing and Spatial Inform. Scien. (ISPRS), Vol.XXXIV, Part 3A, pp.265-270 , ISSN 1682-1750.

15. Sagris, D., Mitsi, S., Bouzakis, K.-D., Mansour, G. (2004). *Geometric design optimization of spatial RR robot manipulator using a hybrid algorithm*, Acta Technica Napocensis, Vol.47(II), pp.717-722, ISSN: 1221-5872.

16. Sagris, D., Mitsi, S., Bouzakis, K.-D., Mansour, G. (2004). *5-DOF robot base location optimization using a hybrid algorithm*, Mecatronica, Vol.1, pp.76-81, ISSN: 1583-7653.

17. Trucco, E., Fusiello, A., Roberto, V. (1999). *Robust motion and correspondence of noisy 3-D point sets with missing data*, Pattern Recognition Letters, Vol.20/9, pp.889-898, ISSN: 0167-8655.

18. Visual Numerics, (1997). *IMSL Fortran subroutines for mathematical applications*.

19. Zhang, Z. (1994). *Iterative point matching for registration of free-form curves and surfaces*, Int. Journal of Computer Vision, Vol.13, pp.119-152, ISSN: 0920-5691

Received: 12 January 2010 / Accepted: 10 May 2010 / Abstract published online: 18 May 2010