Avrupa Bilim ve Teknoloji Dergisi
Özel Sayı 28, S. 1190-1196, Kasım 2021
© Telif hakkı EJOSAT'a aittir
Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
Special Issue 28, pp. 1190-1196, November 2021
Copyright © 2021 EJOSAT
Research Article

Performance Analysis of Steepest Descent-Line Search Condition Combinations in Nonlinear Least Squares Fitting of CMM Data

Kadir Kiran *,1,2

1*Department of Airframe and Power Plant Maintenance, School of Civil Aviation, Suleyman Demirel University, Isparta, Turkey, (ORCID: 0000-0002-6109-435X), kadirkiran@sdu.edu.tr

2Design and Manufacturing Technologies Research Laboratory, Innovative Technologies Application and Research Center, Suleyman Demirel University, Isparta, Turkey

(1st International Conference on Applied Engineering and Natural Sciences ICAENS 2021, November 1-3, 2021)

(**DOI:** 10.31590/ejosat.1012096)

ATIF/REFERENCE: Kiran, K. (2021). Performance Analysis of Steepest Descent-Line Search Condition Combinations in Nonlinear Least Squares Fitting of CMM Data. *European Journal of Science and Technology*, (28), 1190-1196.

Abstract

This paper presents a benchmarking study on the steepest descent (SD) method considering three different line search conditions including Backtracking (BC), Armijo-Backtracking (ABC) and Goldstein (GC) in nonlinear least squares fitting of measured data obtained from coordinate measuring machine (CMM). Within this scope, five primitive geometries such as circle, square, rectangle, triangle and ellipse were built via 3D printer. Those geometries were then scanned with CMM to acquire their 2D profiles to be fitted. To find best fitting parameters for each geometry, the nonlinear least squares approach along with the above-mentioned optimization method-line search condition combinations were employed. During the fitting process, the total number of function evaluations, when the combination converges to required tolerance, were used as a performance metric of the combination in question. With those data, the performance and data profiles for each combination were created to be able to carry out a reliable performance evaluation. Based on those profiles, it has been seen that the SD-ABC combination is the fastest one. In addition, it is successful on all the geometries while the others are not. For the second fastest combination, the SD-BC combination stands out. However, its successful rate is only 80%, which means it fails on a geometry. On the other hand, the SD-GC combination takes the last place in this study. All those results have shown that the line search conditions have a great contribution to the success and performance of the optimization algorithm being used. Besides, their performance may differ from problem-to-problem. The end-users should consider these facts to find best optimization method-line search condition combination for their problems.

Keywords: steepest descent method, line search condition, performance profiles, data profiles, nonlinear least squares fitting, optimization.

Koordinat Ölçme Makinesi Verilerinin Doğrusal Olmayan En Küçük Kareler Uydurulmasında En Dik İniş-Doğru Boyunca Arama Şartı Kombinasyonlarının Performans Analizi

Öz

Bu makale koordinat ölçme makinesinden (KÖM) elde edilen verilerin doğrusal olmayan en küçük kareler uydurulmasında Backtracking (BC), Armijo-Backtracking (ABC) ve Goldstein (GC) içeren üç farklı doğru boyunca arama şartlarını dikkate alarak en dik iniş (EDİ) yöntemi üzerine bir kıyaslama çalışması sunmaktadır. Bu kapsamda, daire, kare, dikdörtgen, üçgen ve elips şekillerindeki beş temel geometri 3B yazıcı ile imal edildi. Daha sonra bu geometrilerin uydurulacak 2B profillerini elde etmek için adı geçen geometriler KÖM ile tarandı. Her bir geometriye en iyi uydurma parametresini bulmak için, doğrusal olmayan en küçük kareler yaklaşımı yukarıda bahsedilen optimizasyon yöntemi-doğru boyunca arama şartı kombinasyonları ile birlikte kullanıldı. Uydurma süreci boyunca ilgili kombinasyon istenilen tolerans değerine yakınsadığında ortaya çıkan toplam fonksiyon değerlendirme sayısı kullanılan kombinasyonun bir performans metriği olarak dikkate alındı. Güvenilir bir performans analizi yapabilmek amacıyla bu veriler ile, her bir kombinasyon için performans ve veri profilleri oluşturuldu. Adı geçen profillere dayanarak EDİ-ABC kombinasyonun en hızlı olduğu görüldü. Ek olarak bu kombinasyon diğer kombinasyonların aksine tüm geometrilerde başarılıdır. İkinci en hızlı kombinasyon için EDİ-BC kombinasyonu ortaya çıkmaktadır. Fakat, adı geçen kombinasyonun başarı oranı sadece %80'dir, yani bir geometride başarısız olmaktadır. Öte yandan, EDİ-GC kombinasyonu bu çalışmada son sırayı almaktadır. Tüm bu sonuçlar gösteriyor ki, doğru boyunca arama şartlarının kullanılan optimizasyon yönteminin başarısına ve performansını büyük bir katkısı vardır. Ayrıca bu şartların performansı problemden probleme farklılık gösterebilir. Son kullanıcılar kendi problemleri için en iyi optimizasyon yöntemi-doğru boyunca arama şartı kombinasyonunu bulmak için bu bulguları dikkate almalıdır.

Anahtar Kelimeler: En dik iniş metodu, Doğru boyunca arama, Performans profilleri, Veri profilleri, Doğrusal olmayan en küçük kareler uydurma, Optimizasyon

[*] Corresponding Author: kadirkiran@sdu.edu.tr

1 Introduction

The steepest descent method, developed by Cauchy [1], along with a line search has been widely implemented in many fields for optimization purpose. As such, in [2], a steepest descent optimization procedure was proposed to evaluate the circularity of a mechanical part. The authors used the geometrical methods for computing the steepest descent direction and step length and they reported that the proposed algorithm is more computationally efficient than the traditional optimization-based and computational geometry-based methods. Xiao et al. [3] employed the SD method for maximum power point tracking and the results showed that it is superior to hill climbing method. An algorithm based on the SD method for blind signal separation was presented by Dam et al. [4]. For optimal step length, they used a combination of Golden search technique and parabolic interpolation and it was determined from the computational experiments that the proposed algorithm provides faster converge than the one which uses constant step length. In addition to this study, regarding as the step length, Kalousek [5] released a random step length selection procedure for the SD method. On the other hand, the crystal lattice parameters were found by implementing the SD method in [6]. Another application of the SD method on the sensor placement problem was completed by Akbarzadeh et al. [7]. Furthermore, an energy minimization approach using the SD method for micromagnetics was presented by Exl et al. [8]. Besides those practical studies, various methods and algorithms, such as these of Refs. [9-11], have been proposed to improve the SD method performance in the literature.

It is concluded from the literature studies summarized above that the SD method is a useful and a practical tool for many applications and its performance may vary depending on the problem dealing with and the step length computations techniques being used. By keeping those facts in mind, in this paper, it is concentrated on performance evaluation of the SD method considering three well-known line search conditions including BC, ABC and GC in the nonlinear least squares geometry fitting. For this purpose, the five test geometries were built with 3D printer and then, their 2D profiles were acquired via the CMM. With implementing an iterative nonlinear least squares geometry fitting procedure, the best parameters, which enable to represent the measured data, for the test geometries have been identified. During the fitting process, the number of function evaluations at each iteration are monitored and the total number of function evaluations when the best parameters are found are set as the performance criteria of the SD-line search condition combination in question. Henceforth, the paper is organized as follows: Section II accommodates the nonlinear fitting procedure, as well as includes test geometries. Section III covers the experimental studies. The obtained results with discussion are provided in Section IV. Finally, Section V summarizes and concludes the paper.

2. Nonlinear Least Squares Fitting Procedure of CMM Data

To obtain the CMM data for nonlinear least squares fitting, the five test geometries containing circle, square, triangle, ellipse and rectangle have been selected. Their parametric mathematical models [12] are given as follows:

Circle:

$$\begin{aligned} x &= r_c \cos(u) + x_c \\ y &= r_c \sin(u) + y_c \end{aligned} \tag{1}$$

Square and rectangle:

$$\begin{aligned} x_u &= \frac{w}{2}(|\cos(u)|\cos(u) + |\sin(u)|\sin(u)) \\ y_u &= \frac{h}{2}(|\cos(u)|\cos(u) - |\sin(u)|\sin(u)) \\ x &= x_u \cos(\theta) - y_u \sin(\theta) + x_c \\ y &= x_u \sin(\theta) + y_u \cos(\theta) + y_c \end{aligned} \tag{2}$$

Triangle:

$$\begin{aligned} r &= \frac{h}{cos\left(\frac{2}{n_s} arcsin\left(sin\left(\frac{n_s}{2}u\right)\right)\right)} \\ x_u &= r\, cos(u) \\ y_u &= r\, sin(u) \\ x &= x_u\, cos(\theta) - y_u\, sin(\theta) + x_c \\ y &= x_u\, sin(\theta) + y_u\, cos(\theta) + y_c \end{aligned} \tag{3}$$

Ellipse:

$$\begin{aligned} x_u &= a \cos(u) \\ y_u &= b \sin(u) \\ x &= x_u \cos(\theta) - y_u \sin(\theta) + x_c \\ y &= x_u \sin(\theta) + y_u \cos(\theta) + y_c \end{aligned} \tag{4}$$

In these equations, x and y are the geometry final coordinates, x_u and y_u are the geometry coordinates without rotation, x_c and y_c are the center coordinates of the geometry, u is the parameter ranging from 0 to 2π radians, r_c is the radius of circle, w is the width of square and rectangle, h is the height of square, rectangle and triangle, θ is the rotation angle, r is the radius of triangle, n_s is the number of edges (i.e., $n_s = 3$ for triangle), a is the radius of ellipse along the x axis and b is the radius of ellipse along the y axis.

It is well-established in the literature that the nonlinear least squares fitting method aims to minimize the sum of squared error between measured and computed data. To adapt this approach for geometry fitting, the following parameter dependent sum of the squared error [13], $\epsilon^2(p)$, is defined.

$$\begin{aligned} \epsilon^2(p) = &\sum_{j=1}^{n}\left[x_j - x_j^{model}(p)\right]^2 \\ &+ \sum_{j=1}^{n}\left[y_j - y_j^{model}(p)\right]^2 \end{aligned} \tag{5}$$

In Eq. (5), p is the parameter vector of the geometry (i.e., $p = [r_c\ x_c\ y_c\ u]$ for circle, $p = [w\ h\ x_c\ y_c\ \theta\ u]$ for square and rectangle, $p = [h\ x_c\ y_c\ \theta\ u]$ for triangle and $p = [a\ b\ x_c\ y_c\ \theta\ u]$ for ellipse), n is the number of data points, x_j and y_j are the coordinates acquired via CMM, x_j^{model} and y_j^{model} are the coordinated computed via mathematical model. To find the best parameter vector, which minimizes the $\epsilon^2(p)$, a line search procedure is used as follows:

$$p_{i+1} = p_i + \alpha h \qquad (6)$$

where $h = -J^T D$ is the search direction which is the steepest descent direction in this study, J is the Jacobian matrix of the objective function and $D = \begin{bmatrix} D_x \\ D_y \end{bmatrix}$, $D_x = x_i - x_i^{model}(p)$ $D_y = y_i - y_i^{model}(p)$. On the other hand, $\alpha > 0$ is the step length that defines the amount of the movement along the steepest descent direction. There are several computation methods (i.e., line search conditions) for this step length in the literature. We use three well-known line search conditions (i.e., BC, ABC and GC) [14] in this study. Their mathematical descriptions are given in Table 1.

Table 1. Line search conditions

BC	$\epsilon(p_i + \alpha h) \leq \epsilon(p_i)$
ABC	$\epsilon(p_i + \alpha h) \leq \epsilon(p_i) + \zeta \alpha G^T h$
GC	$\epsilon(p_i) + (1 - \nu)\alpha G^T h \leq \epsilon(p_i + \alpha h) \leq \epsilon(p_i) + \eta \alpha G^T h$

In Table 1 $G = J^T D$ is the gradient of the objective function, ζ and η are the scalars. $\zeta = 0.25$ and $\eta = 0.35$ are used in here. The line search procedure with calculated step length at each iteration (i.e., Eq. (6)) keeps computing parameter vector until the converge condition is met, which is:

$$maximum|J^T D| \leq 10^{-3} \qquad (7)$$

3. Experimental Studies

The geometries given in the previous section have been first built with 3D printer using PLA material. Their 2D profiles have been then obtained via the CMM, as shown in Fig. 1.

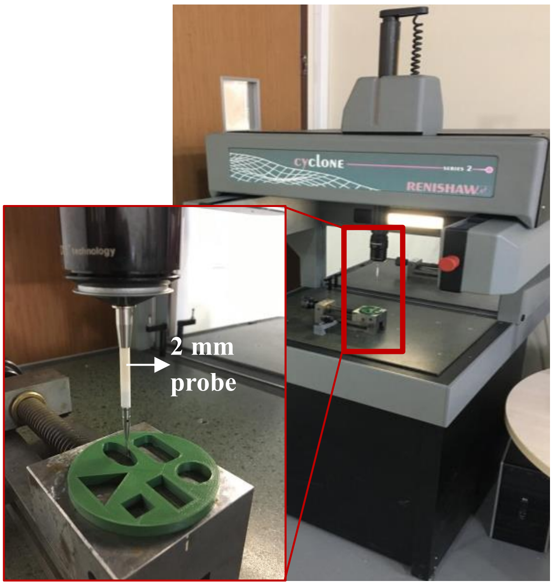

Fig. 1 Scanning of geometries

The scanning has been performed with 2 mm touch prob and the scanning speed $V_s = 100$ mm/min was used. An example CMM measurement for a square with this setup are shown in Fig. 2. As mentioned before, the collected data (i.e., x_j and y_j) are employed as the input for Eq. (5).

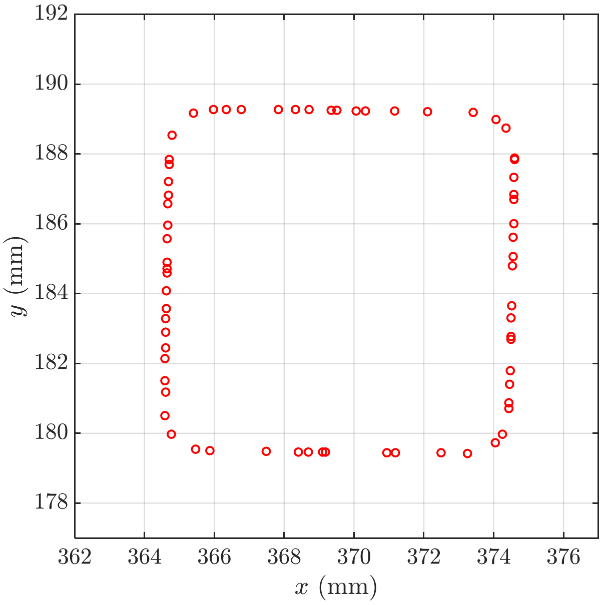

Fig. 2 Example CMM measurement

4. Results and Discussion

To test the effect of the line search conditions on the SD method performance, the nonlinear least squares geometry fitting procedure has been completed for all the geometries using the SD-line search condition combinations (i.e., SD-BC, SD-ABC and SD-GC). While performing this procedure, all the geometry parameters are monitored and recorded for every iteration. Besides, the sum of least square error, norm of the objective function gradient and the number of function evaluations to compute the step length are recorded to check the fitting procedure health. As an example for those data, Fig. 3 shows parameters and step length progresses at each iteration for a square fitting using the SD-ABC combination. In addition, the sum of least squares error and norm of the objective function gradient progresses are illustrated in Fig. 4.

As can be seen from Fig. 3, all the parameters successfully converge after a little bit fluctuation. Notice also that the step length α varies at each iteration. This is due to line search conditions the have to be met for a remarkable progress. Otherwise (i.e., using constant step length), it might require much more iterations to converge or, the algorithm may fail to complete fitting process. It noteworthy that although the use of varying step lengths may be beneficial for significant progress in the parameter search, they also bring computational cost, which is core of the paper. This will be elaborated in later of this section. However, to contribute this fact, a plot showing the number of function evaluations at each iteration are provided in Fig. 5. As seen, a quite amount of work to compute the step length is required to consume because the number of parameters to be found are large. More specifically, for this example square fitting, the number of parameters (i.e., $p = [w\ h\ x_c\ y_c\ \theta\ u]$) to be found is 72. This large parameter value is emerged from the u which should be correspond to the number of measured points (i.e., 67 points in this fitting).

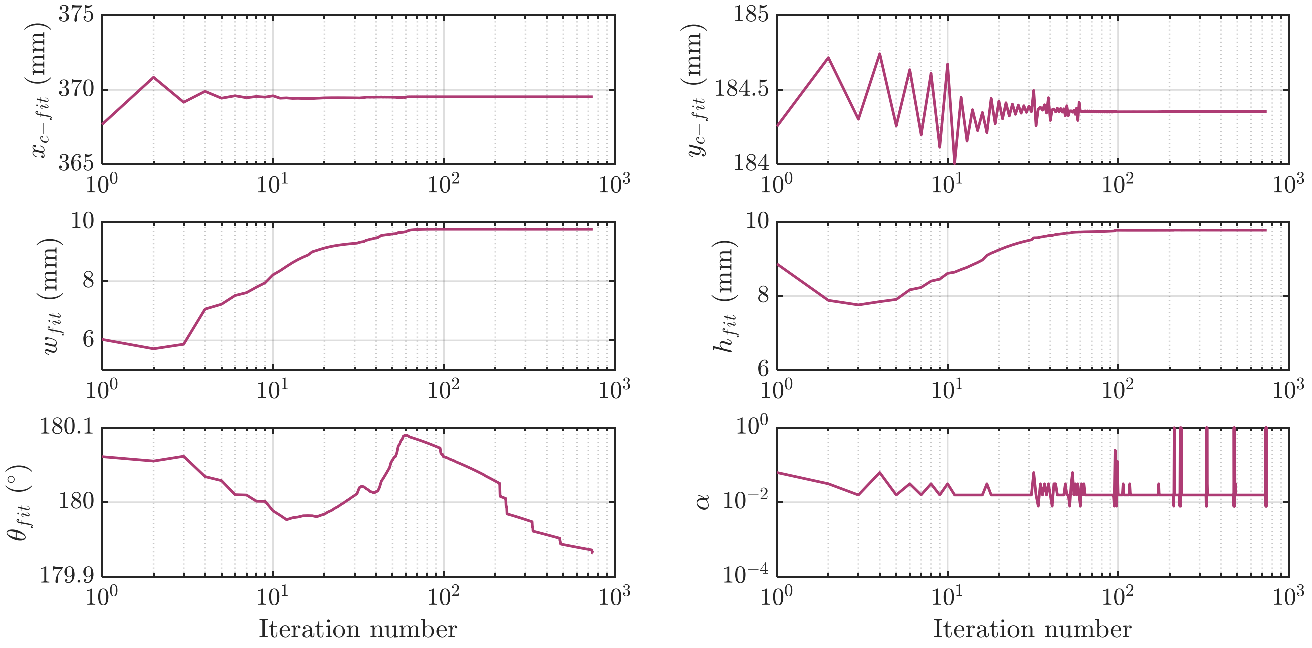

Fig. 3 Fitting parameters and step lengths

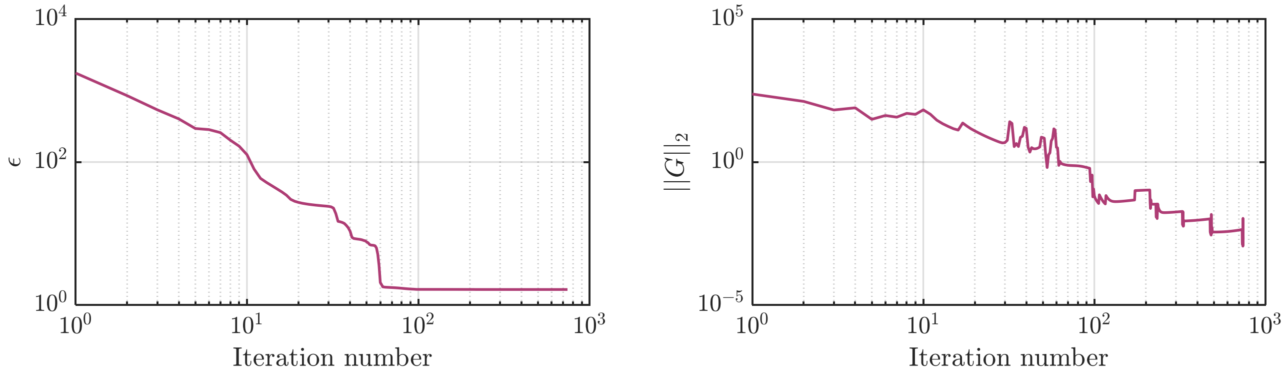

Fig. 4 Sum of least square error and norm of objective function gradient progresses

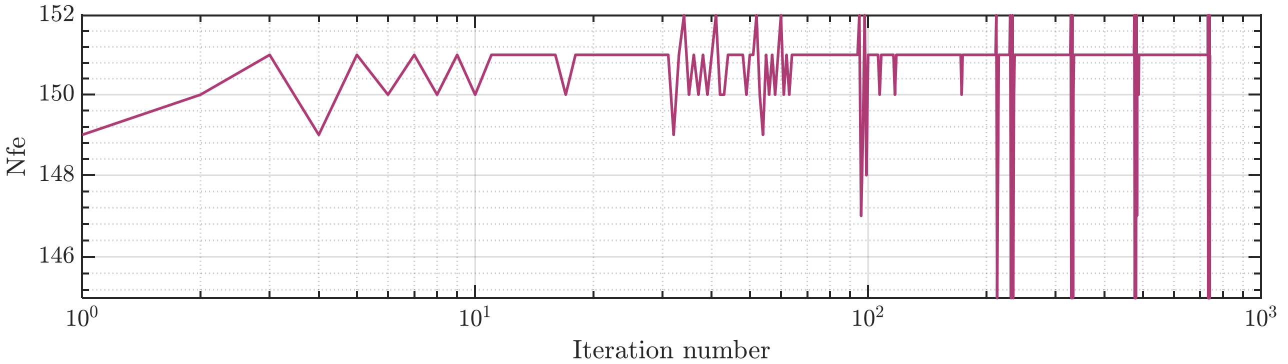

Fig. 5 Number of function evaluations at each iteration during the fitting process

The reason behind defining the u as a parameter is the irregular spacing between the measured points (see Fig. 2). Otherwise, from personal experience, the algorithm fails to complete geometry fitting process. This fact makes the geometry fitting more attractive to work on.

As a result of the progresses shown in Figs. 3 to 5, we achieved a successful square fit, as shown in Fig. 6. This figure also provides a comparison of actual, fit and measured squares. As seen, the square fit is satisfyingly able to represent the measured data. From the practical application of view, this fit, for instance, enables us to identify the part tolerances, manufacturing accuracy (e.g., both additive and traditional manufacturing processes), quality inspection, tool path compensation amount etc. In Fig. 6, one can notice that there is a slight difference between the actual, fit and measured squares. The actual square dimensions are defined as $w = h = 10$ mm in the design step. However, the fit performed on measured data gives the dimensions to be $w = 9.7580$ mm, $h = 9.7812$ mm.

The difference between the actual and fit dimensions shows the manufacturing tolerance that might vary depending on the geometry to be built and the material to be used in the 3D printing.

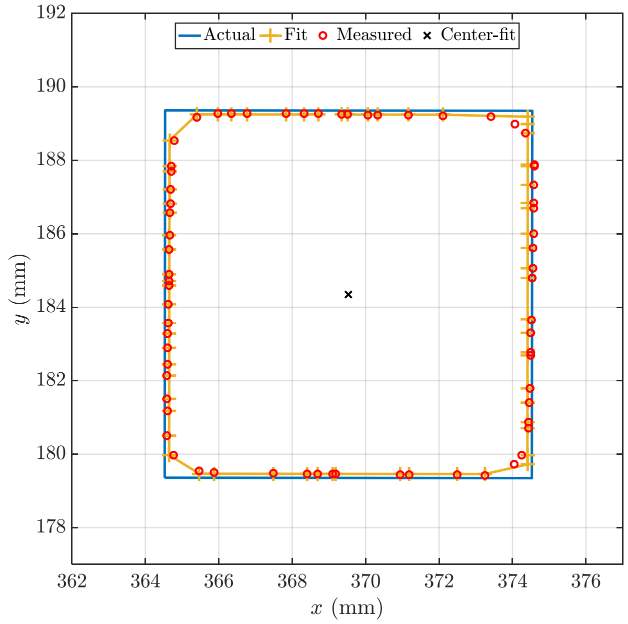

Fig. 6 A comparison between actual, fitted and measured squares

In other respect, the same parameter monitoring and fit quality checking procedures reported so far are also carried out in all the geometry fittings with the SD-line search condition combinations. When the combination in question meets converge requirement (i.e., Eq. 7), the line search stops and the total number of function evaluations performed thus far is recorded to be a performance criterion of the combination. By doing so, Figs. 7 is obtained.

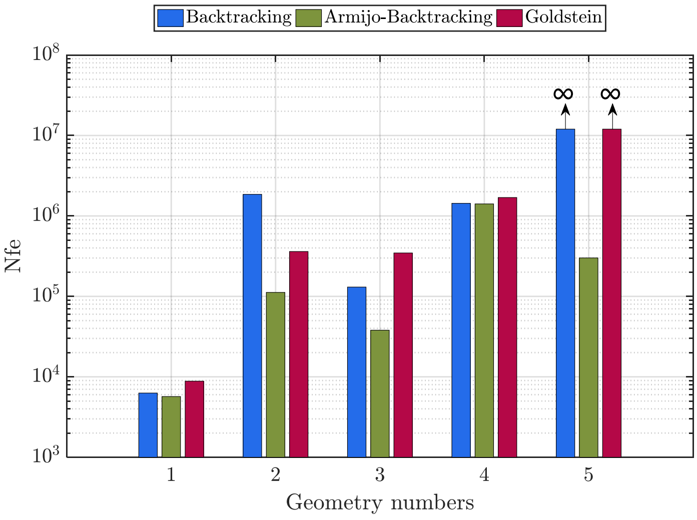

Fig. 7 Total number of function evaluations

This figure shows the total number of function evaluations for each line search condition, corresponding the geometries, when the converge occurs. In the figure, the geometry numbers (i.e., 1, 2, 3, 4 and 5) represent the circle, square, triangle, ellipse and rectangle, respectively. Besides, ∞ denotes that the corresponding SD-line search conditions combination is not successful to find the geometry parameters. Specifically, the SD-BC and SD-GC combinations could not complete the fitting process, thereby their total number of function evaluations are set to infinity. By focusing on the figure, one can realize that the SD-ABC combination performs well-done on all the geometries and it requires the lowest number of function evaluations for converging compared to others. In addition, the circle geometry seems the easiest one to fit because all the combinations are successful on it with lower number of function evaluations in comparison with other geometries. This is attributed that the circle geometry equation is simpler and it does not need any rotation around its center while others require (see Eqs. 1 to 5). Similarly, we can state that the hardest one to fit is rectangle geometry due to the fact that the SD-BC and SD-GC combinations fails. Analogue observations can be extended by just looking at the Fig. 7. However, for more probabilistic analysis, we move on to the performance profiles, as shown in Fig. 8.

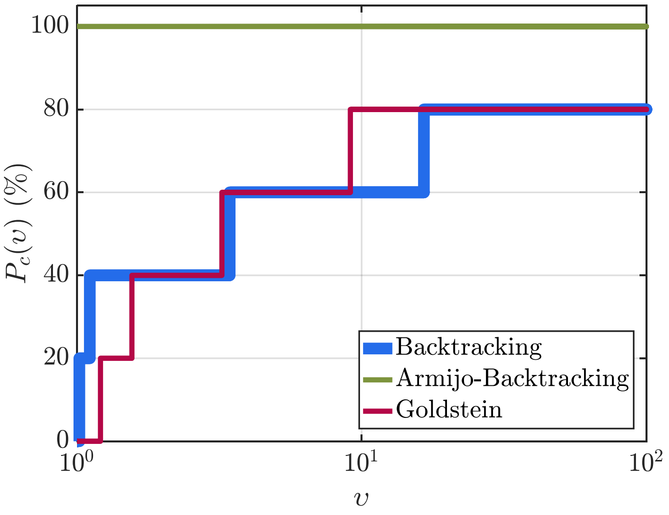

Fig. 8 Performance profiles-1

The idea behind the performance profiles, developed by Dolan and More [15], is to give the success probability of the solver on the test set within the given factor. By adapting this approach in the current study, we achieve the performance profiles of the line search conditions. They provide efficient and reliable assessments on the line search conditions. For instance, by looking at the values of $P_c(\upsilon = 1)$, we can determine the fastest line search condition which is the ABC. As seen from Fig. 8, it is the fastest one on all the geometries because its success probability at the factor $\upsilon = 1$ is $P_c(\upsilon = 1) = 100\%$ while others are 0% . This observation confirms the previous one conducted from Fig.7. As increasing υ, the success probability of BC and GC start rising. Within the factor $\upsilon = 1.11$ of the fastest condition, for instance, the BC is able to be successful on the 2 geometries (i.e., $P_c(\upsilon = 1.11) = 40\%$). The BC success rate remains same at the factor range $1.11 \leq \upsilon \leq 3.43$. This range for the GC is $1.56 \leq \upsilon \leq 3.22$. At the factor $\upsilon = 9.13$ of the fastest condition, the GC reaches maximum success rate (i.e., $P_c(\upsilon = 9.13) = 80\%$) whereas the BC requires $\upsilon = 16.55$ for maximum performance. For clarity, $P_c(\upsilon) = 80\%$ means that the conditions in question are able to be successful on the 4 geometries out of 5. Fig. 7 verifies this fact as mentioned before. On the other hand, for identifying the second fastest condition, we need to exclude the first one because the performance profiles are dependent each other. By excluding the ABC and generating the performance profiles with rest of them, we obtain Fig. 9. As have been done before, it is possible to determine the second fastest one based on the $P_c(\upsilon = 1)$ values. It is obvious from Fig.9 that the second fastest one is the BC with the probability $P_c(\upsilon = 1) = 60\%$.

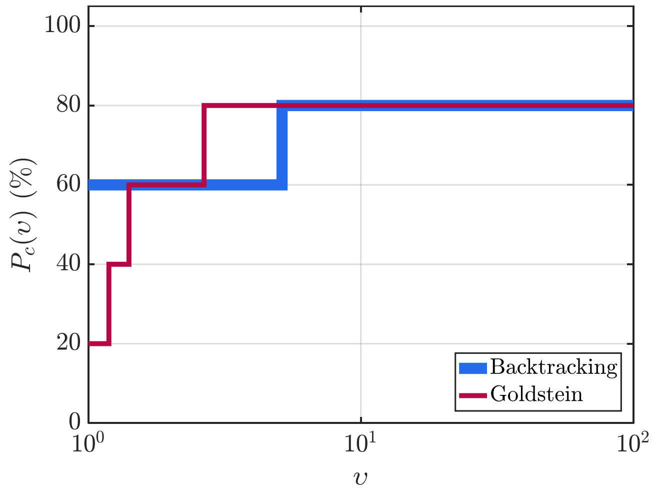

Fig. 9 Performance profiles-2

In other words, the BC has higher number of wins compared to GC (i.e., $P_c(\upsilon = 1) = 20\%$). Based on Fig.7, we can confirm this observation, which the BC has lower number of function evaluations on the geometries 1, 3 and 4 (i.e., circle, triangle and ellipse, respectively) than the GC. This property of the performance profiles is quite useful for speed ranking of the line search conditions. However, it is sometimes given a computational budget in terms of algorithm running time, memory usage, number of function evaluations, etc. and it is required to solve the certain number of problems within this budget. In this case, the performance profiles cannot be employed due to their dependency to each other. For an independent evaluation from a computational budget (i.e., total number of function evaluations in this study) perspective, we use the method of data profiles [16]. As such, the data profiles for all the line search conditions are indicated in Fig. 10 within the total number of function evaluations range $10^3 \leq \psi \leq 10^7$. In this figure, $D(\psi)$ denotes the percentage of the geometries, which are fitted, within the given computational budget ψ. To illustrate, for $\psi = 10^4$ all the line search conditions are able to complete fitting of only one geometry that is the circle (i.e., $D(\psi) = 20\%$). If we increase the ψ to 10^5 the ABC fits one more geometry, which is the triangle (i.e., geometry number 3, see Fig. 7), while others remain same. The speed of the SD-ABC combination also appears in data profiles as expected.

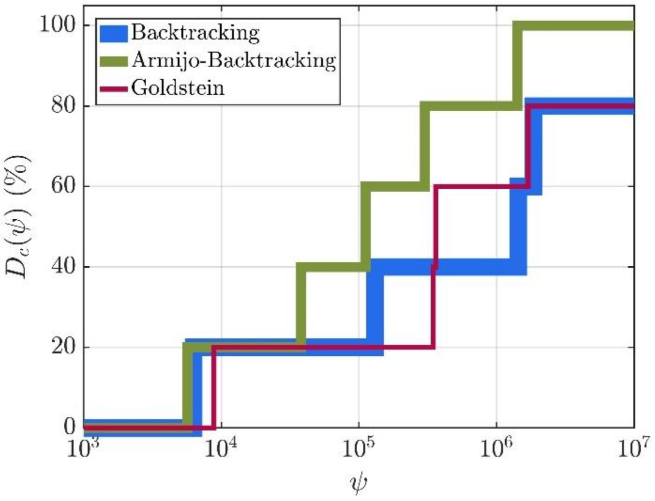

Fig. 10 Data profiles

By combining the knowledges obtained from performance and data profiles on the line search conditions, it is obvious that the ABC is the fastest and most robustness (i.e., fits all the geometries).

5. Conclusions

This paper has been concentered on the performance assessment of the SD-line search condition combinations in nonlinear least squares geometry fitting. The three combinations, containing the SD-BC, SD-ABC and SD-GC, performances have been evaluated on the five geometries. Their 2D profiles were obtained using the CMM. In geometry fitting process, all the geometry parameters were kept track and the total number of function evaluations when the converge occurs were used as the performance measure of the combination in question. Using this measure, the performance and data profiles were generated for efficient performance assessments. It was seen that the geometry fitting is not a straightforward task because the u parameter for each measured point must be defined as a search parameter. It means that all the u parameters of measured points have to be included to be variables in addition to the number of dimension and rotation variables. This makes search parameter vector p large enough and difficult to fit. On the other hand, it was observed that the line search conditions play a crucial role for the SD method performance and success. For a remarkable performance, the end-users should consider this fact and it may be recommended to use the SD-ABC combination, which was found to be fastest and most robustness in this study, for geometry fitting. Moreover, the line search condition parameters (i.e., ζ and η for the ABC and GC, respectively) might have an effect on the performance of the SD method, which deserves further explorations.

Acknowledgment

The author acknowledges Design and Manufacturing Technologies Research Laboratory, Innovative Technologies Application and Research Center, Suleyman Demirel University where the experimental studies were performed.

References

[1] Cauchy, A. (1847). Methode generale pour la resolution des systemes d'equations simultanees. *Comp. Rend. Sci. Paris,* 25(2), 536-538.

[2] Zhu, L. M., Ding, H., Xiong, Y. L. (2003). A steepest descent algorithm for circularity evaluation. *Computer-Aided Design,* 35, 255-265.

[3] Xiao, W., Dunford, W. G., Palmer, P. R., Capel, A. (2007). Application of centered differentiation and steepest descent to maximum power point tracking. *IEEE Transactıons on Industrıal Electronıcs,* 54, 2539-2549.

[4] Dam, H. H., Nordholm, S., Low, S. Y., Cantoni, A. (2007). Blind signal separation using steepest descent method. *IEEE Transactıons on Sıgnal Processıng,* 55, 4198-4207.

[5] Kalousek, Z. (2017). Steepest descent method with random step lengths. *Found Comput Math,* 17, 359-422.

[6] Shirokanev, A. S., Kirsh, D.V., Kupriyanov, A.V. (2016). Applıcatıon of gradıent steepest descent method to the problem of crystal lattice parametric identification. *In CEUR Workshop Proceedings,* 1638, 393-400.

[7] Akbarzadeh, V., Lévesque, J. C., Gagné, C., Parizeau, M. (2014). Efficient sensor placement optimization using gradient descent and probabilistic coverage. *Sensors,* 14, 15525-15552.

[8] Exl, L., Bance, S., Reichel, F., Schrefl, T., H. Stimming, P., Mauser, N. J. (2014). LaBonte's method revisited: An effective steepest descent method for micromagnetic energy minimization. *Journal of applied physics,* 115, 17D118.

[9] Andrei, N. (2006). An acceleration of gradient descent algorithm with backtracking for unconstrained Optimization. *Numer Algor,* 42, 63-73.

[10] Quiroz, E. A. P., Quispe, E. M., Oliveira, P. R. (2008). Steepest descent method with a generalized Armijo search for quasiconvex functions on Riemannian manifolds. *J. Math. Anal. Appl.,* 341, 467-477.

[11] Samir, C., Absil, P. A., Srivastava, A., Klassen, E. (2012). A gradient-descent method for curve fitting on riemannian manifolds. *Found Comput Math,* 12, 49-73.

[12] https://www.desmos.com (Access date:16.05.2021).

[13] Jia, P. (2017). Fitting a parametric model to a cloud of points via optimization methods. Ph.D. thesis, Syracuse University, New York, USA.

[14] Nocedal, J., Wright, S. J. (2006). *Numerical optimization*, 2nd ed., New York, USA: Springer Science & Business Media.

[15] Dolan, E. D., More, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical programming,* 91(2), 201-213.

[16] More, J. J., Wild, S. M. (2009). Benchmarking derivative-free optimization algorithms. *SIAM Journal on Optimization,* 20(1), 172-19.