



POLITECNICO
MILANO 1863

DIPARTIMENTO DI MECCANICA



Comparison of chaos optimization functions for performance improvement of fitting of non-linear geometries

MORONI, GIOVANNI; SYAM, WAHYUDIN PERMANA; PETRO', STEFANO

This is a post-peer-review, pre-copyedit version of an article published in Measurement. The final authenticated version is available online at:

<http://dx.doi.org/10.1016/j.measurement.2016.02.045>

This content is provided under [CC BY-NC-ND 4.0](https://creativecommons.org/licenses/by-nc-nd/4.0/) license



Comparison of chaos optimization functions for performance improvement of fitting of non-linear geometries

Giovanni Moroni ^{1,2}, Wahyudin P. Syam ¹, and Stefano Petrò ¹

¹ Mechanical Engineering Department, Politecnico di Milano, Via La Masa 1, 20156 Milan, Italy

² School of Mechanical Engineering, Tongji University, Cao An Road, 201804 Shanghai, P. R. China

Abstract

Fitting algorithm plays an important role in the whole measuring cycle in order to derive a measurement result. It involves associating substitute geometry to a point cloud obtained by an instrument. This situation is more difficult in the case of non-linear geometry fitting since iterative method should be used. This article addresses this problem. Three geometries are selected as relevant case studies: circle, sphere and cylinder. This selection is based on their frequent use in real applications; for example, cylinder is a relevant geometry of an assembly feature such as pin-hole relationship, and spherical geometry is often found as reference geometry in high precision artifacts and mechanisms.

In this article, the use of Chaos optimization (CO) to improve the initial solution to feed the iterative Levenberg-Marquardt (LM) algorithm to fit non-linear geometries is considered. The results of the use of CO to improve the initial solution show that higher quality fitting results can be obtained in term of smaller norm of the residuals, while preserving the computational cost. Moreover, fitting an “incomplete-point-cloud”, i.e. the situation where the point cloud does not cover the whole feature (e.g. covers an hemisphere only), is presented as well. As core of this article, comparison study of the efficiency of different one-dimensional maps of CO is presented. This study shows that, in general, logistic-map function provides the best solution compared to other types of one-dimensional functions. Finally, case studies on hemispheres and industrial cylinders fitting are presented.

Keywords: Least-square fitting, non-linear optimization, chaos optimization, one dimensional map.

1. Introduction

Geometrical metrology has an important role in manufacturing. It is the procedure to verify geometric attributes of products and is most often realized by coordinate metrology [1]. In coordinate metrology, least-square (LS) fitting to associate substitute geometry to point cloud is a fundamental step to derive the measurement result [2-5]. By applying this procedure, dimensional measurements, e.g. the diameter of a circle or sphere, the angle between two lines, the distance between two spheres, etc, and geometrical measurements, e.g. flatness, roundness, parallelism, coaxiality deviations, etc are performed accurately and flexibly. Therefore, the fitting algorithm is a fundamental element in the whole chain of dimensional metrology. Figure 1

presents the role of fitting in the measurement cycle. Based on Hoppe et. al. [6], fitting the substitute geometry is called a “function reconstruction”, since it is an association of points to a defined function. From the point of view of the fitting algorithm, basic geometries can be divided in two groups: linear and non-linear. Linear geometries include 2D line and plane. Most other types of basic geometries, such as circle, sphere, cylinder, and cone are considered non-linear geometries.

The fitting of linear geometries can be analytically solved and a single global optimal result obtained. In the case of non-linear geometries fitting, an iterative procedure, minimizing some objective function, is used instead. The lack of prior knowledge about the parameters of the geometry to fit significantly increases the difficulties in the fitting itself. Consequently, for non-linear geometric fitting an initial solution must be selected to initialize the iterative procedure. This initial solution should be good enough to ease the search algorithm in the estimation of the best fitting geometry. The situation in which the prior knowledge about the geometry to be fitted is not known can be found in reverse modeling. One example is that it is very difficult to identify the axis of direction of a cylinder when prior knowledge about the cylinder is unknown. Not only accuracy problem in the fitting process but also cost of computation is relevant for this situation. The reason is that current measurement instruments, especially the non-contact one, are able to capture thousands or even millions of points within short period of time. As such, time required for the fitting becomes relevant issues as it increases the overall measuring time. Hence, fast and accurate fitting is needed to realize a high-speed inspection to reduce inspection cost, and hence reducing the production cost [7]. Hence, not only an accurate, but also a high-speed fitting procedure is required.

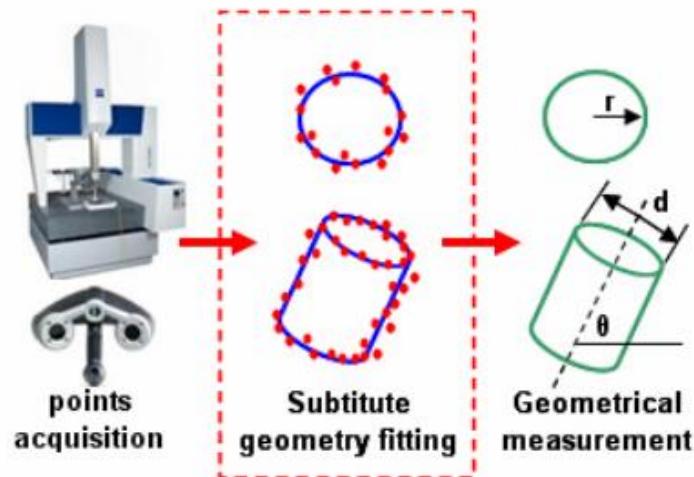


Figure 1: Role of geometric fitting in a metrology cycle.

In a previous paper [8] the authors presented the efficiency of non-linear fitting by utilizing CO method to select the initial point for the LM iterative algorithm (Chaos-LM algorithm). Results of the study shows that CO improves the fitting accuracy without scarifying the computation time [8]. In this paper, comparison among CO functions for the LS fitting of non-linear geometries by Chaos-LM algorithm are presented. The purpose is to study the performance of CO method with respect to its different types of continuous one-dimensional map functions in different fitting situation and to identify the most effective for most cases. The paper is structured

as follows. Introduction of non-linear geometries and their distance function is presented in section 2. In section 3, non-linear fitting method and its characteristic is described. In this section, the problem of initial solution for the iterative procedure is presented. CO optimization and its various one-dimensional functions are explained in section 4. The performance improvement by using CO method to determine the initial solution, comparison among one-dimensional map functions and case studies are presented in section 5. Finally, concluding remark and future developments are proposed in section 6.

2. Non-linear Geometry

The defining parameters of geometries define them as either linear or non-linear. Line and Plane are classified as linear geometry since their defining parameters are linear. Other basic geometries have non-linear parameters which define their shapes. They are circle, sphere, cylinder, cone and torus. Therefore, they are named “non-linear geometries”. This paper addresses the problem of non-linear Least Squares (LS) fitting circle, sphere, and cylinder. Circular and spherical geometries have many applications, for example the sphere, as it univocally defines a single point (its center) is often adopted as reference geometry [9-10] and also commonly adopted as artifact the verification of the performance of metrology instruments [11-12]. Many rotational part found in mechanical product are constituted by circular features such as shafts and holes. Moreover, a pin-hole is the most common assembly feature, and it is represented by a couple of mating cylinders [13].

To fit the LS feature to a cloud of point means to associate a feature the cloud of points; the associated feature minimizes the sum of the squared orthogonal distances from the points of the cloud to the feature itself (fig 2). The sum of squared orthogonal distance is then the objective function to minimize usually by means of an iterative algorithm:

$$\mathbf{a} = \arg \min_{\mathbf{a}} \sum_{i=1}^n d^2(\mathbf{p}_i, \mathbf{a}) \quad (1)$$

Where: \mathbf{p}_i is the i th point of the cloud, n is the number of points, \mathbf{a} is a vector of parameters defining a geometrical feature (e.g. the coordinates of the center and the radius of a sphere), and $d(\mathbf{p}_i, \mathbf{a})$ represents the orthogonal distance of \mathbf{p}_i from the geometrical feature defined by the parameters \mathbf{a} . For the circle in a plane, the distance function is (Fig. 2a left):

$$d(x_i, y_i, x_0, y_0, r) = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2} - r \quad (2)$$

Where x_0, y_0 are the coordinates of the circle center, r is the circle radius, and x_i, y_i are the coordinates of the i th point.

For the sphere, the objective distance function can be similarly formulated as:

$$d(x_i, y_i, z_i, x_0, y_0, z_0, r) = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2} - r \quad (3)$$

The distance function of a point to a cylinder is more complex compared to circle and sphere ones (see fig. 2a right). It is defined as:

$$d(\mathbf{p}_i, \mathbf{p}_0, \mathbf{n}, r) = \|(\mathbf{p}_i - \mathbf{p}_0) \times \mathbf{n}\| - r \quad (4)$$

where r is the radius of the cylinder, \mathbf{p}_0 is a point on the axis of the cylinder, and \mathbf{n} is the versor of the cylinder axis (fig. 2b).

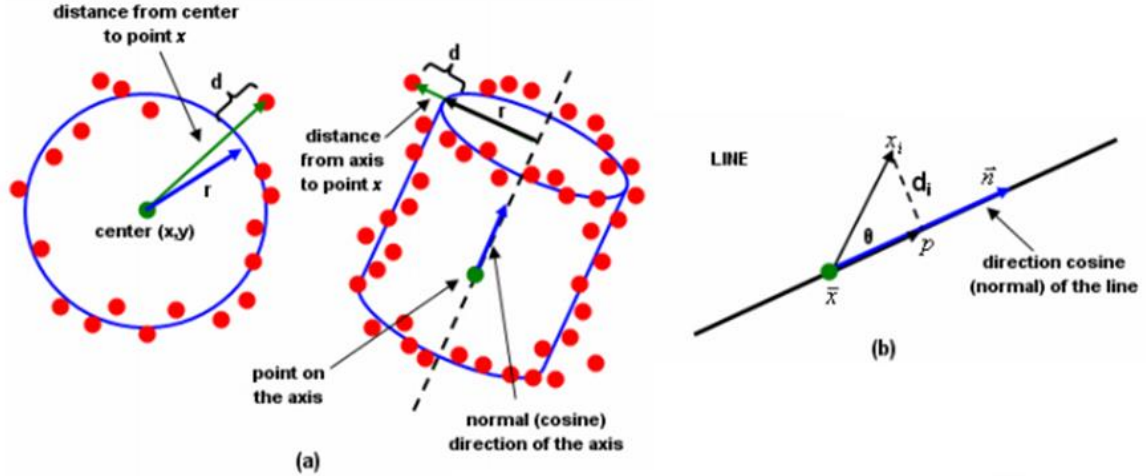


Figure 2: (a) Definition of point distance for circle (sphere) and cylinder, (b) Definition of point distance for 3D-line.

One can observe that the objective function to minimize is a non-linear multi-modal function. This multi modal function has many local minima and/or maxima.

3. Non-linear fitting

As the proposed geometries are non-linear, a non-linear fitting algorithm will be required to solve their associated non-linear LS problems. Many non-linear LS algorithm are known in literature, among which probably one of the most often implemented and adopted is the Levenberg-Marquardt algorithm.

3.1 Levenberg-Marquardt Algorithm.

The well-known Levenberg-Marquardt (LM) algorithm can be used to solve non-linear least square problems [14-15]. There are two methods applied by LM algorithm, the first one is steepest-descent (gradient search) and the second one is Gauss-Newton step. Firstly, steepest-descent approach is carried out to quickly reach the optimum area. When the search is already around the optimum region, the approach is changed to Gauss-Newton one to search the optimum solution within a smaller optimum region. Steepest-descent search is to minimize a function. It searches through the direction of the gradient of the function. Let F be the function to optimize, and x_k the candidate solution at step k . For minimization case, the next iteration step of a steepest-descent algorithm is:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \lambda_s \nabla F(\mathbf{x}_k) \quad (5)$$

Where $\nabla F(\mathbf{x})$ is the gradient of the objective function calculated at \mathbf{x} , and λ_s is the step size, which determines the distance of the next candidate solution from the current one. Hence, if λ_s is set very small, it will take long to reach convergence. Otherwise, the optimum value will be over-looked if it is set too large.

In the Gauss-Newton method instead, a linearization of the non-linear function is applied. A Taylor expansion series is deployed for this linearization procedure. The series is:

$$\nabla F(\mathbf{x}) = \nabla F(\mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0)^T \nabla^2 F(\mathbf{x}_0) + \underbrace{\dots + \left[(\mathbf{x} - \mathbf{x}_0)^T \right]^n \nabla^n F(\mathbf{x}_0)}_{\text{Higher order terms}} \quad (6)$$

With reference to eq. 6, only the linear term is considered. It has two reasons. Firstly, the algorithm is more efficient to reach the convergence. Secondly, the form is tractable to solve \mathbf{p} . By setting $\nabla f(\mathbf{p}) = 0$, the next iteration of the Gauss-Newton is described as:

$$\mathbf{p}_{j+1} = \mathbf{p}_j - (\mathbf{J}_d^T \mathbf{J}_d)^{-1} \mathbf{J}_d^T \mathbf{d}(\mathbf{p}_j) \quad (7)$$

Where $\mathbf{d}(\mathbf{p}_j)$ is the vector of the residual (distances) at step j , and \mathbf{J}_d is the Jacobian matrix of this vector of distance functions. Since, only the linear term is considered in the Taylor series, the accuracy of this linearization is only valid within a very small region (or interval). This region is called trust region. Due to this reason, Gauss-Newton method is valid for the use within small search region; else it tends to get stuck in local minima. Hence, based on this characteristic, the method is effective when starting near the optimum solution.

Combination between steepest-decent and Gauss-Newton methods is the advantage of LM method. A vector of input parameters \mathbf{p}_0 , which are parameters which define the shape of geometry to be fitted, is supplied to the LM algorithm, along with a matrix \mathbf{M} which is a $n \times 3$ matrix. The matrix contains data points, defined as: $\mathbf{M} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$ where $\mathbf{x} = [x_i, y_i, z_i]^T$. The goal is to get \mathbf{p} which is the optimized vector. The LM method used here is based on the LM used by NIST [5] for their reference algorithm. The LM algorithm is:

Algorithm 1: Levenberg-Marquardt Algorithm

Input: Vector \mathbf{p}_0 of the initial guess for the parameter and matrix \mathbf{M} containing the points to fit.

Output: Vector \mathbf{p} of the optimized parameter

- 1: Set $\lambda = 0.0001$
- 2: DO { decrease λ
- 3: set $\mathbf{U} = \mathbf{J}_0^T \mathbf{J}_0$
- 4: set $\mathbf{v} = \mathbf{J}_0^T \mathbf{d}(\mathbf{p}_0)$
- 5: set $F_0 = \mathbf{d}^T(\mathbf{p}_0) \mathbf{d}(\mathbf{p}_0)$
- 6: DO { increase λ

```

7:      set  $\mathbf{H} = \mathbf{U} + \lambda(\mathbf{I} + \text{diag}(\mathbf{U}))$ 
8:      solve  $\mathbf{H}\mathbf{x} = -\mathbf{v}$ 
      set
9:       $\mathbf{p}_{new} = \mathbf{p}_0 + \mathbf{x}; \text{set } F = \mathbf{d}^T(\mathbf{p}_{new})\mathbf{d}(\mathbf{p}_{new})$ 
10:     IF converged THEN return  $\mathbf{p}_0 = \mathbf{p}_{new}$ 
11:     UNTIL  $F_{new} < F_0$  or stop criterion is true
12:     IF  $F_{new} < F_0$  THEN  $\mathbf{p}_0 = \mathbf{p}_{new}$ 
13:     UNTIL stop criterion is true

```

The parameter λ is the LM variable. This variable is set based on NIST method [5]. \mathbf{J}_0 is a Jacobian matrix, so that its i -th row is $\nabla d_i(\mathbf{p}_0)$. For the circle, the parameters \mathbf{p}_0 are x_0, y_0 (the coordinate of the center) and the radius r . For the sphere, similarly to circle, only one additional element z_0 coordinate of the center needs to be added. Finally, the parameters for the cylinder are x_0, y_0, z_0 , i.e. the coordinates of a point on the axis, its versor $\mathbf{n} = [n_1, n_2, n_3]$, and finally its radius r . \mathbf{J}_0 is a $n \times m$ matrix in which the number of column m corresponds to the number of parameters to estimate, and the number of rows n corresponds to number of points to fit.

The main LM method can be observed in step 7 ($\mathbf{H}\mathbf{x} = -\mathbf{v}$). If this equation is enlarged into $\mathbf{J}_0^T \mathbf{J}_0 + \lambda(\mathbf{I} + \text{diag}(\mathbf{J}_0^T \mathbf{J}_0))\mathbf{x} = -\mathbf{J}_0^T \mathbf{d}(\mathbf{p}_0)$, it can be observed that if λ is zero or small, the LM algorithm becomes a Gauss-Newton method. On the other hand, if λ is large enough, then the off-diagonal elements of $\mathbf{J}_0^T \mathbf{J}_0$ will have less effect, and it becomes the steepest-descent method. The term $\mathbf{I} + \text{diag}(\mathbf{J}_0^T \mathbf{J}_0)$ is which is a weighted distance based on a Nash [15] suggestion, so that $\mathbf{H}\mathbf{x} = \mathbf{J}_0^T \mathbf{J}_0 + \lambda(\mathbf{I} + \text{diag}(\mathbf{J}_0^T \mathbf{J}_0))\mathbf{x} = -\mathbf{J}_0^T \mathbf{d}(\mathbf{p}_0)$ becomes positive definite. In this situation, the calculation of the matrices is numerically more stable.

3.2. Problem of seeding initial point

The description of the LM algorithm shows clearly that a un initial solution \mathbf{p}_0 must be provided [16]. The choice of the initial solution is critical, because the function to optimize is multi-modal: many local optima and high curvature characterize it. Therefore, the search can get trapped locally in a non-global optimum. Fig. 3(left) shows examples of this type of function by using Schweifel equation and the square of the summation of a circle distance function, which is $F = \sum_{i=1}^N d_i^2$, where d_i is formulated in eq. (2).

As mentioned before, the fitness function of least squares fitting is multi-modal, so the optimization algorithm can get trapped in a local optimum. The probability of getting trapped highly depends on the starting solution. [16]. Figure 3 (right) is a good illustration to explain the phenomena. It is an objective function to fit a circle. Different candidate radius r can be observed as different level of surface (color) and the surface is obtained by varying the (x, y) which is the center location of a circle. From the figure, it is clear that the searching process during the

optimization step can be trapped in one of the surface region. Therefore, this cause an sub-optimal solutions are likely to be found. Figure 4 illustrates how the starting solution directly affects the final results. An unexpected sub-optimal solution is found when the initial guess is far from optimum (figure 4a). On the other hand, the final solution is significantly improved by using a good initial guess (yielding a lower fitness function value, fig. 4b).

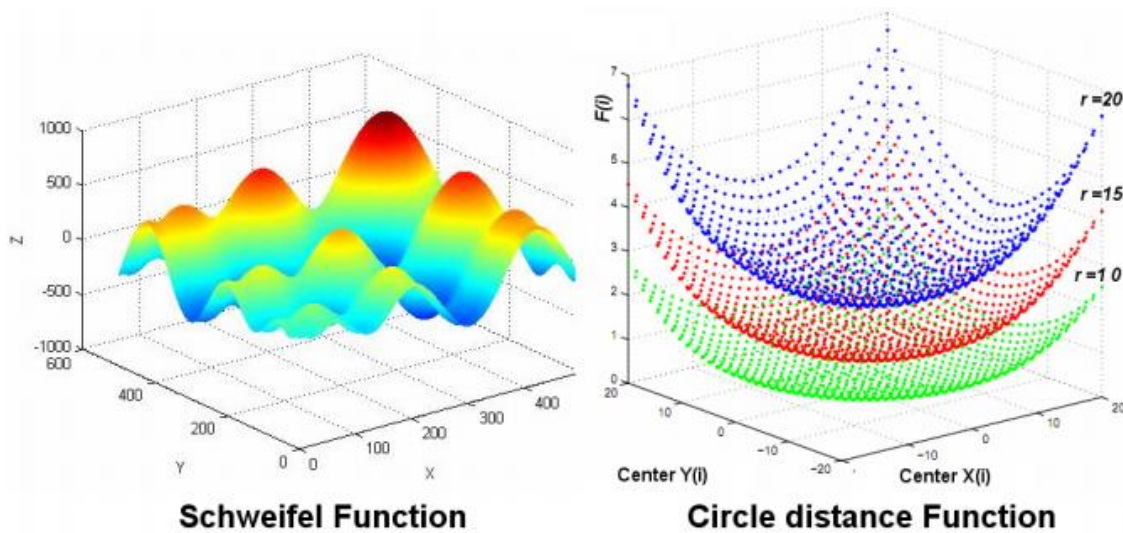


Figure 3: Plots of Schweifel function and squared circle distance.

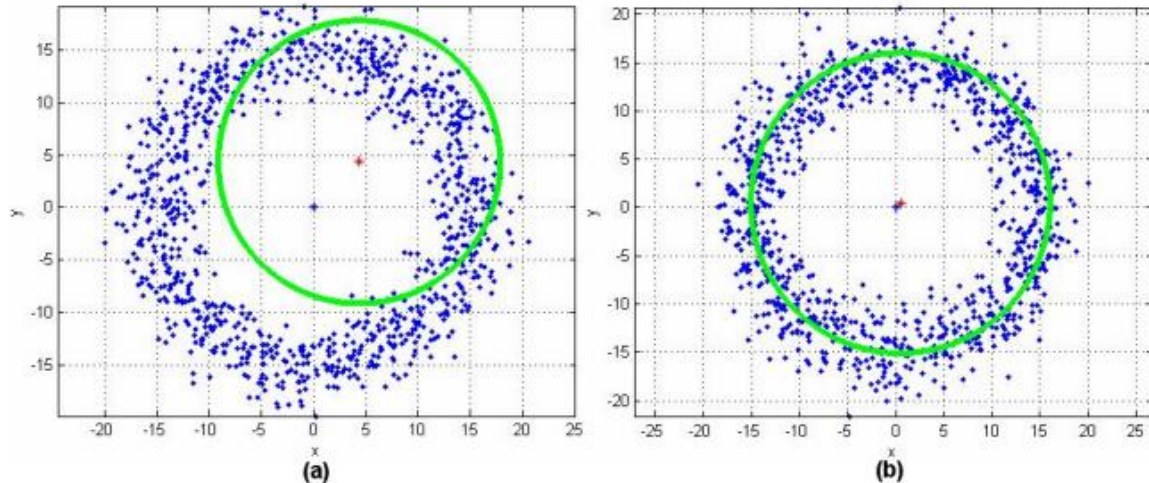


Figure 4: Effect of initial solution to the final fitting result. (a) Initial guess is far from optimal, (b) initial guess is near optimal.

4. Chaos Optimization

4.1 Chaos algorithm

Chaos is a semi-random behavior generated by a non-linear function. It creates a chaotic dynamic step which can be useful to escape from local optima region in a search process. It is

deterministic because each step can be uniquely determined from the previous step; however, though deterministic, the step will realize a hardly predictable path. Hence, it is similar to the observation of a stochastic process. The concept differs from the improved heuristic searches (random-based algorithm) which work based on rejection-accepting probability test [17]. Since chaos dynamic is not stochastic, it differs from heuristic search. Searching through regularity of chaotic motion is its fundamental recipe [18]. This motion represents a dynamical trajectory system motion that can be represented as mapping of one variable to other variable. Chaos function significantly depends on its initial condition in the sense that two initial values, which can be spatially very close to each other, when subject to chaotic step trajectories will diverge at exponential rate. Considering a mapping of $\mathbf{t}_{k+1} = \mathbf{F}_M(\mathbf{t}_k)$, which is n -dimensional variables, the significant difference of the chaos trajectory \mathbf{t}_{k+1} with respect to two initial conditions \mathbf{t}_0 and $\mathbf{t}_{0+\Delta}$, can be modeled by a Lyapunov exponent as [21]:

$$\left| \mathbf{F}_M^n(\mathbf{t}_0) - \mathbf{F}_M^n(\mathbf{t}_{0+\Delta}) \right| = \Delta e^{n\lambda(\mathbf{t}_0)} \quad (8)$$

where n is number of iteration and the Lyapunov exponent is represented by the $\lambda(\mathbf{t}_0)$ function. A function, to have a chaotic behavior, should have dimension ≥ 3 [26]. Otherwise, this behavior can be observed in one-dimensional functions if the map is not invertible. A map \mathbf{F}_M is not invertible, if and only if, given \mathbf{t}_{k+1} , we cannot solve $\mathbf{t}_{k+1} = \mathbf{F}_M(\mathbf{t}_k)$ for \mathbf{t}_k . In this case, the solution of $\mathbf{t}_k = \mathbf{F}_M^{-1}(\mathbf{t}_{k+1})$ does not exist. This is due to one single value of \mathbf{t}_{k+1} can be mapped to more than one values of \mathbf{t}_k . Chaotic orbit properties, which are ergodicity, stochastic property, and regularity are used by Chaos optimization (CO) [19] to guide the searching process inside the optimization region. For example, one common dimensional map function is the logistic map, which is formulated as:

$$\mathbf{t}_{k+1} = \lambda_c \mathbf{t}_k (1 - \mathbf{t}_k) \quad (9)$$

Where k is iteration number and $\lambda_c \in \{3.56, 4\}$ is a control argument. Value of $0 \leq \mathbf{t}_0 \leq 1$ where $\mathbf{t}_0 \notin \{0, 0.25, 0.5, 0.75, 1.0\}$ is selected based on Yang's recommendation [20]. Equation (9) becomes chaotic. Chaotic here means that its value is drastically changed within the limit of λ_c and \mathbf{t}_k . This is the representation of the regularity of chaotic motion. Plot of time series of this function and paired-plot between two consecutive chaos variables are shown in figure 5. From this figure, one can observe that with a small change in two consecutive t , a chaotic behavior (spread value) will be observed in the time series manner (figure 5 left).

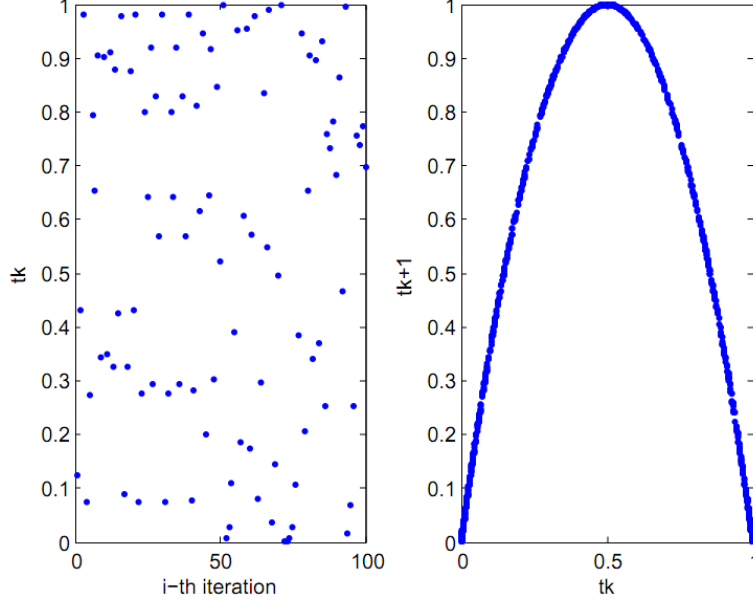


Figure 5: Plot of Logistic map one-dimensional function. (left) time-series plot of the logistic map, (right) paired-plot between two consecutive chaos variables t .

Based on the chaotic characteristic, to improve the initial guess of LM non-linear fitting, the CO method is used. Moreover, the computation time will be preserved. This is important for situation in which the sample size is large (millions of points). The Chaos-LM algorithm is presented as follow:

Algorithm 2: Chaos search to improve the initial guess in LM method (Chaos-LM Algorithm)

Input: Vector \mathbf{p}_0 of the initial guess for the parameter (1:n-param)

Goal: New vector \mathbf{p}_0 is the improved initial guess for

$\text{Min } F(\mathbf{p}_i)$, $\mathbf{p}_i \in \{\mathbf{a}_i, \mathbf{b}_i\}$, Let $\mathbf{p}^k = (\mathbf{p}^1 : \mathbf{p}^k)$, $\mathbf{t}^k = (\mathbf{t}^1 : \mathbf{t}^k)$

where: $\mathbf{p} = (p_1, \dots, p_n)$ and $\mathbf{t} = (t_1, \dots, t_n)$

1: Set $k = 0, r = 0$, Set $k_{\max} = 10, r_{\max} = 30$

2: Produce randomly. $\mathbf{t}_0 \in \{0,1\}$ and $\notin \{0,0.25,0.5,0.75,1.0\}$.

Set $\mathbf{t}^k = \mathbf{t}^0, \mathbf{t}^* = \mathbf{t}^0, \mathbf{a}^0 = \mathbf{p} - \text{MPE}, \mathbf{b}^0 = \mathbf{p} + \text{MPE}$

3: where: $\mathbf{a} = (a_1, \dots, a_n)$, $\mathbf{b} = (b_1, \dots, b_n)$

Note: MPE is maximum permissible error of the instrument

4: Set $\mathbf{p}^* = \mathbf{p}_0 \rightarrow$ **initial guess parameter**

DO WHILE { $r < r_{\max}$;

5: DO WHILE { $k < k_{\max}$;

Set $p_i = a_i^r + t_i^r (b_i^r - a_i^r)$;

6: calculate $F^k = \sum_{i=0}^N d_i^2(p_i)$

IF $F^k < F^*$ THEN

7: $F^* = F(p^k), p^* = p^k, t^* = t^k$

8: $k = k + 1$;

Generate Chaotic motion for the next iteration, for example,

using logistic one-dimensional map function:

$$t_i^k = \lambda t_i^{k-1} (1 - t_i^{k-1}), \lambda \in \{3.56, 4\}$$

9: }END k -th iteration; $r = r + 1$
 $a_i^{r+1} = p_i * -\gamma(b_i^r - a_i^r)$

10: and
 $b_i^{r+1} = p_i * +\gamma(b_i^r - a_i^r)$

11: IF $a_i^{r+1} > a_i^r$ THEN $a_i^{r+1} = a_i^r$, $\gamma \in \{0, 0.5\}$

12: IF $b_i^{r+1} < b_i^r$ THEN $b_i^{r+1} = b_i^r$, $\gamma \in \{0, 0.5\}$

13: IF $r < r_{\max}$ THEN produce $t_0 \in \{0, 1\}$ by random,
 $k = 0$, $\mathbf{t}^k = \mathbf{t}^0$ GOTO(7)
 ELSE CO is terminated,

14: return $\mathbf{p}_0 = \mathbf{p}^*$;}
 END r -th iteration;

15: Insert the new p_0 into **Algorithm 1: LM algorithm**.

There are parameters to be defined for the CO algorithm. In this case for logistic map chaotic motion generator, the parameters are set as $\gamma = 0.45$ [19], $\lambda = 4$ [20], $r_{\max} = 50$, and $K_{\max} = 50$. The reason of this parameter selection is that to adjust small ergodic range around p_i^* . For a long term chaotic behavior from a small change of t , the value $\lambda = 4$ is set to be equal to 4. The purpose is that this additional step does not contribute significantly to the computation time. The statements IF $a_i^{r+1} > a_i^r$ THEN $a_i^{r+1} = a_i^r$, $\lambda \in \{0, 0.5\}$ and IF $b_i^{r+1} < b_i^r$ THEN $b_i^{r+1} = b_i^r$, $\lambda \in \{0, 0.5\}$ spread the search movement farther from the initial bounding area which have been set in the beginning of the search, so that it will increase the probability of finding the global optimum region. With reference to Figure 4 (right), the final result will be much improved if the initial guess is expected to lie near the correct optimization zone.

4.2 One-dimensional map functions

The chaotic motion generator is calculated from one-dimensional deterministic functions. There are several types of continuous one-dimensional map functions to generate chaotic motions to explore the search space of feasible solutions [21-25], even though the most general type is the logistic map [21], which has been introduced in §4.1. These different types of one-dimensional maps can work efficiently on different kind problems. Therefore, the comparison between one-dimensional functions can find the one characterized by the best performance. In the following, several types of one-dimensional maps will be compared to identify the optimal one for initialized the LM algorithm to solve the LS fitting problem. Both clouds covering the whole surface of the feature and partial clouds will be considered in the comparison. The next few paragraphs introduce the considered maps.

4.2.1 Tent map

This type of map is characterized by a triangle shape for \mathbf{t}_{k+1} with respect to \mathbf{t}_k [21] and can be formulated as:

$$t_{k+1} = \begin{cases} \mu t_k & t_k < 0.5 \\ \mu(1 - t_k) & t_k \geq 0.5 \end{cases} \quad (10)$$

In this type of map, $\forall t_k \in [0,1]$, the value we have will be $t_{k+1} \in [0,1]$ too. The value of μ is suggested to be set to 2 [21]. Figure 6 shows characteristic of Tent map.

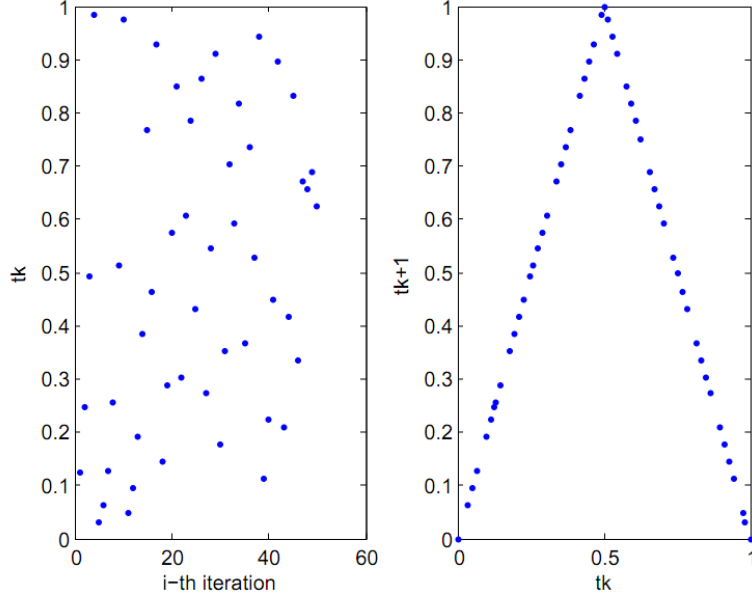


Figure 6: Plot of Tent one-dimensional map function. (left) time-series plot of the Tent map, (right) paired-plot between two consecutive chaos variables t .

4.2.2 Bernoulli shift map

The model comes from the effort to model the packages traffic in a network system [22]. The characteristic of this one-dimensional map is that in any successive iteration, considering two different initial values t_0 and $t_{0+\Delta}$, the trajectories always diverge. The plot of this one-dimensional map characteristic is presented in figure 7. The map can be formulated as:

$$t_{k+1} = \begin{cases} \frac{t_k}{1-\lambda} & 0 < t_k \leq 1-\lambda \\ \frac{t_k - (1-\lambda)}{\lambda} & 1-\lambda < t_k \leq 1 \end{cases} \quad (11)$$

The value of λ is usually set to 0.5 [22].

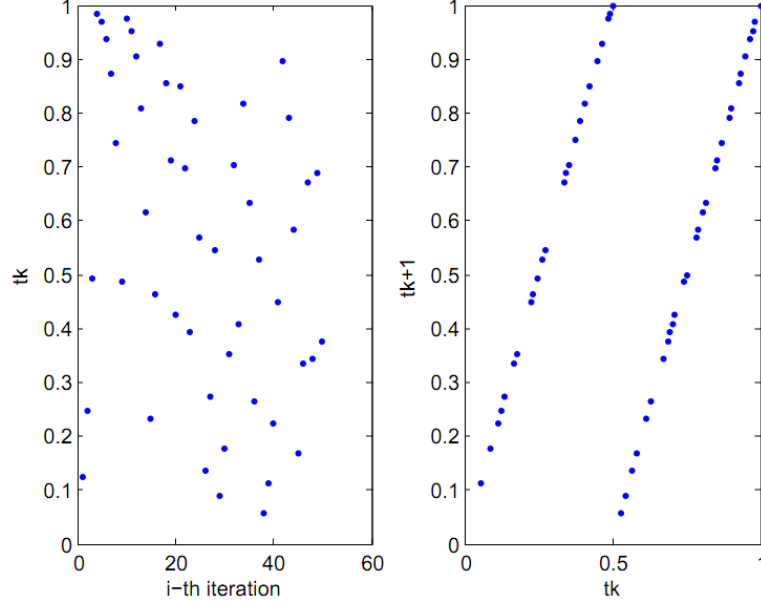


Figure 7: Plot of Bernoulli Shift map one-dimensional function. (left) time-series plot of the Bernoulli map, (right) paired-plot between two consecutive chaos variables t .

4.2.3 Liebovitch map

Liebovitch and Tooth [26] proposed a one-dimensional map to model the kinetic activities of an ion channel. This map consists of three piece-wise linear segments which represent active, passive and switching region. The three segments have interval in $[0,1]$ and they do not overlap.

The Liebovitch one-dimensional map is formulated as:

$$t_{k+1} = \begin{cases} \alpha_1 t_k & 0 < t_k \leq d_1 \\ \frac{d_2 - t_k}{d_2 - d_1} & d_1 < t_k \leq d_2 \\ 1 - \alpha_2(1 - t_k) & d_2 < t_k \leq 1 \end{cases} \quad (12)$$

where $d_1, d_2 \in (0,1), d_1 < d_2$, $\alpha_1 = \frac{d_2}{d_1}(1 - (d_2 - d_1))$, and $\alpha_2 = \frac{1}{d_2 - 1}((d_2 - 1) - d_1(d_2 - d_1))$.

Variables α_1, α_2 determine the behavior of t_k to produce the “dwell time” of the three states. A minimum “dwell time” inside the switching region is preferable to determine the next step of the trajectory by increasing the value of d_2 to be close to 1. Figure 8 shows the linear segment as property of Liebovitch map.

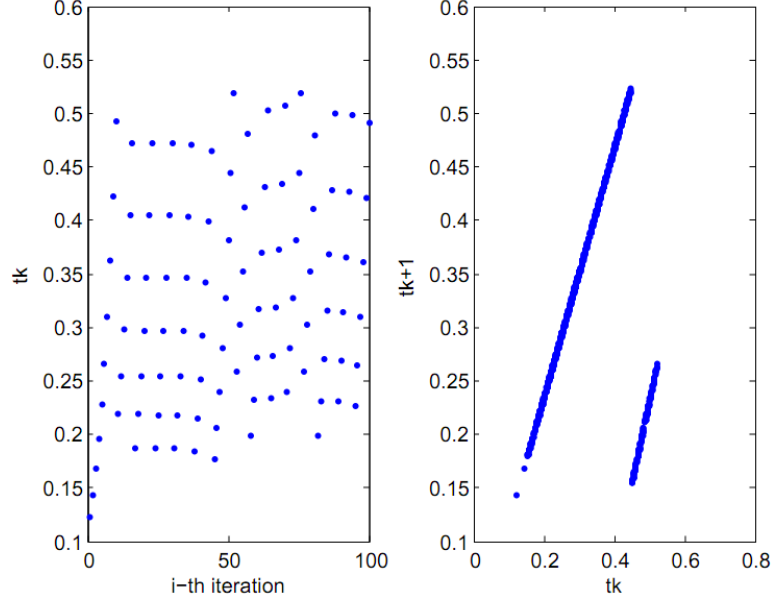


Figure 8: Plot of Liebovitch map one-dimensional function. (left) time-series plot of the Liebovitch map, (right) paired-plot between two consecutive chaos variables t .

4.2.4 Intermittency map

The one-dimensional intermittency map models the intermittency phenomenon, which is an irregular alternation of phases, in turbulence studies [27]. Figure 9 plots this type of one-dimensional map. A “sifting” phenomenon can be observed in its time series plot. This is the extension of Bernoulli map by introducing non-linear piece-wise functions. The two piece-wise functions represent passive and active period, for the first and second piece-wise function, respectively.

$$t_{k+1} = \begin{cases} \varepsilon + t_k + ct_k^m & 0 < t_k \leq d \\ \frac{t_k - d}{1 - d} & d < t_k \leq 1 \end{cases} \quad (13)$$

where $c = \frac{1 - \varepsilon - d}{d^m}$ and ε is set to a very small fraction of real number. In fact, a small ε value increases the time scale, and then the search region [22].

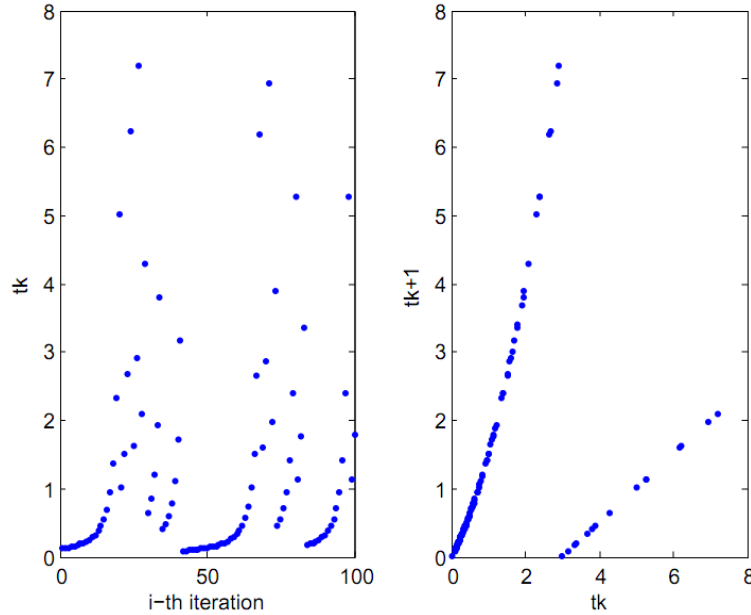


Figure 9: Plot of Intermittency map one-dimensional function. (left) time-series plot of the Intermittency map, (right) paired-plot between two consecutive chaos variables t .

4.2.5 Chebyshev map

Characteristic of this one-dimensional map is that it has infinite collapse (high dynamic) within two symmetrical region of $[-1,0) \cup (0,1]$ [23]. It means that the function is stable on the center point 0 and starts to collapse in the regions far from zero but in $[-1,1]$. The Chebyshev map function is defined as:

$$t_{k+1} = \cos(k \cos^{-1} t_k) \quad (14)$$

The value of k explains the rank of the Chebyshev function to extend the Lyapunov exponent. It is selected equal to 4 [23]. Time series plot and two consecutive variable plot of this map can be observed in figure 10.

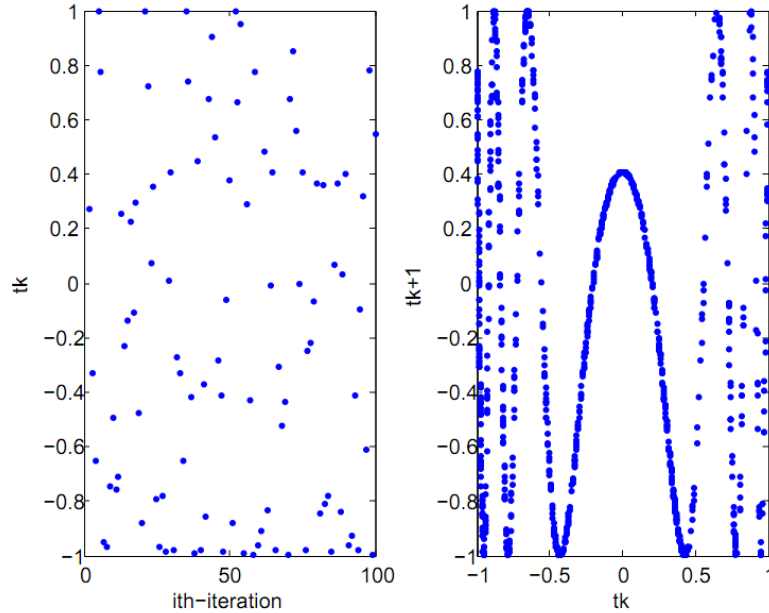


Figure 10: Plot of Chebyshev map one-dimensional function. (left) time-series plot of the Chebyshev map, (right) paired-plot between two consecutive chaos variables t .

4.2.6 ICMIC map

Iterative chaotic map with infinite collapse (ICMIC) is a one-dimensional map similar to the Chebyshev one. The difference between ICMIC and Chebyshev map is that in ICMIC, the function response collapses (is unstable) at the center of two symmetrical region of $[-1,0) \cup (0,1]$. The regions far from the center area are stable area [23]. ICMIC map is defined as:

$$t_{k+1} = \sin\left(\frac{a}{t_k}\right) \quad (15)$$

The value of a is selected equal to 2 so that the Lyapunov exponent number is less than 2 [28] Figure 11 presents the behavior of the ICMIC one-dimensional map. It can be considered as the inverse of the Chebyshev map function.

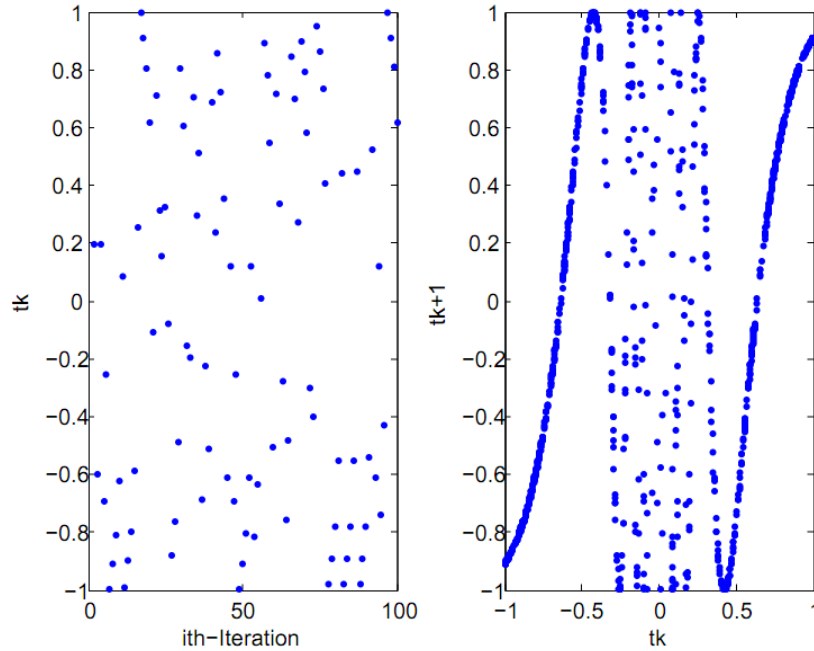


Figure 11: Plot of ICMIC map one-dimensional function. (left) time-series plot of the ICMIC map, (right) paired-plot between two consecutive chaos variables t .

4.2.7 Gaussian map

This one dimensional map has the classical “bell shape” of the Gaussian distribution [24]. The map can be formulated as:

$$t_{k+1} = \exp(-\alpha x_n^2) + \beta \quad (16)$$

where α and β are real numbers. Usually $\alpha = 4.9$ and $\beta = -0.58$ [24]. Figure 12 shows the time-series plot and two consecutive variables plot for the Gaussian one-dimensional map.

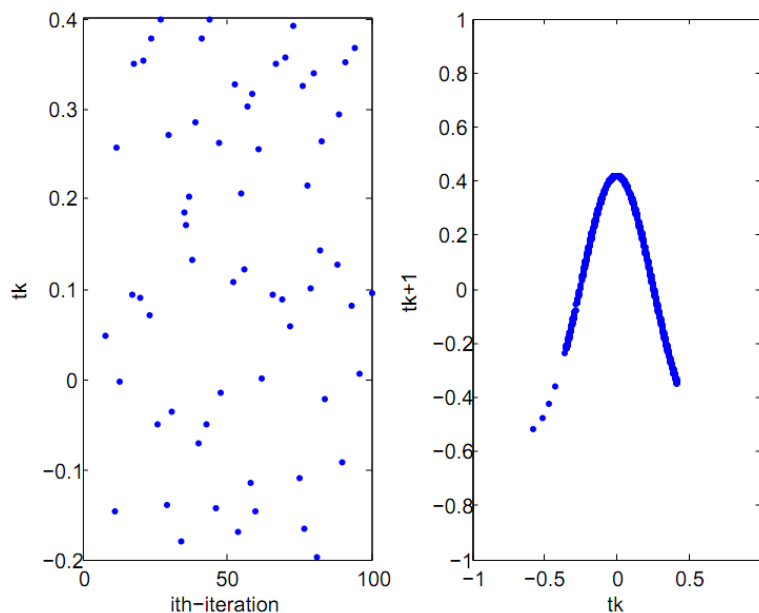


Figure 12: Plot of Gaussian map one-dimensional function. (left) time-series plot of the Gaussian map, (right) paired-plot between two consecutive chaos variables t .

4.2.8 Sine map

The shape of the sine one-dimensional map is qualitatively similar to the logistic map [25]. The formulation of this map is:

$$t_{k+1} = \frac{a}{4} \sin(\pi t_k) \quad (17)$$

where a is an integer number and is selected equal to 2 [25]. This map is shown in figure 13.

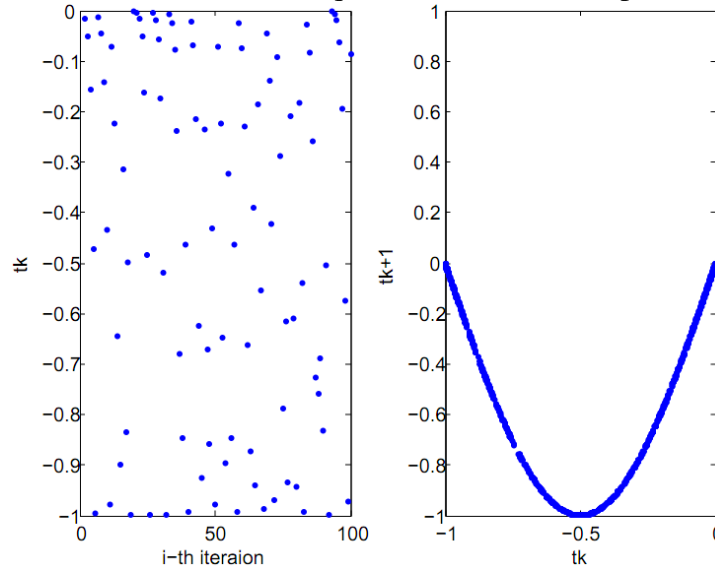


Figure 13: Plot of sine map one-dimensional function. (left) time-series plot of the sine map, (right) paired-plot between two consecutive chaos variables t .

4.2.9 Circle map

Physical motivation of this one-dimensional map the simulation of the behavior of driven mechanical rotors. Andrey Kolmogorov [25] first proposed this. Another physical relevance of this model is that it describes a model of phase locked loop in electronics. The circle map is formulated as:

$$t_{k+1} = t_k + \Omega - \frac{K}{2\pi} \sin(2\pi t_k) \quad (18)$$

where K and Ω are a predetermined constants. K is an integer number and Ω is a real number. Ω represents the external frequency applied to the system, and K is the degree of non-linearity. The main characteristic of this map is that it has more chaotic dynamics with respect to change on its parameters value (figure 14).

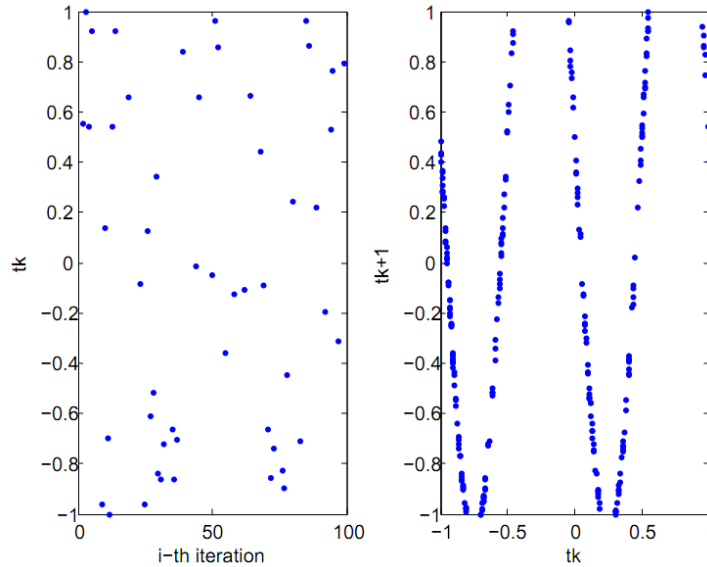


Figure 14: Plot of circle map one-dimensional function. (left) time-series plot of the circle map, (right) paired-plot between two consecutive chaos variables t .

4.2.10 Algorithm complexity

For the combined Chaos-LM algorithm, a complexity analysis of the algorithm is applied to understand the relation between the growths of computational cost with respect to the number of input variables. In this case, the input variable is the number of points n to which the substitute geometry shall be fitted. One can observe that for each algorithm, CO and LM, there are two nested loops. In fact, these loops are not related to the number of points n , but only depends on a defined constant number. Only the procedure to calculate the objective function length relates to n is. As such, the computational growth can be stated as follow. Let the total order of the algorithms be $f(n) = n_1 + n_2 = n + n = 2n$, where subscript 1 and 2 correspond to algorithm 1 (CO) and 2 (LM), respectively. Hence, the algorithm efficiency is $O(n)$ since $\exists n_0 > 0$ and $k > 0 \Rightarrow \forall n > n_0, f(n) \leq k.g(n)$ so that $f(n) = n_1 + n_2 = O(n)$. Hence, the complexity of the algorithm is linear with n .

5. Implementation and Performance comparison

In this section, implementation of the combination of CO and LM algorithm is presented. Following the implementation, the performance comparisons among various function of one-dimensional map are performed. Comparison will consider two kind of case studies: case studies based on the simulation of the sampling of geometric features (simulation approach), and a cased based on the use of real data from the sampling of a physical specimen (real measurement approach). The performance indicators for the fitting algorithm are the magnitude of residual error $\|\mathbf{r}\|$ (where \mathbf{r} is a vector containing the residuals of the fitting), for fitting accuracy, and the computational time, for fitting efficiency. Smaller values of these indicators denote a better performance.

The first approach generates points from some ideal geometry, which can be a circle, a sphere and or a cylinder. Noise is added which can be generated uniformly or normally distributed. Table 1 presents detail on the simulation. There are two levels of standard deviation considered for the data. Type 1 represents only the noise contribution from the instrument. Its sigma value is obtained from the instruments Maximum Permissible Error (MPE) [9-10] Type 2 instead simulates the contribution due to the part to part variability, including the instrument. Points generated by the second type represent a more realistic situation since an inspected part always contains some feature deviation from the nominal geometry [28]. Geometries will be generated both as full geometries and half geometries (e.g. a sphere and an hemisphere), giving rise to a total of six considered geometries. Reason to consider a half-geometry is that in many cases, points obtained by an instrument cannot cover the whole geometry due to, for example, surface accessibility problems, incomplete feature geometry, etc.

Table 1: Details of data generation.

Type of Data		Number of points and Nominal Parameters		
		Circle	Sphere	Cylinder
Uniform	Range [μm]	$(x,y,r)=(15,15,20)$ mm	$(x,y,z,r)=(15,15,15,20)$ mm	Cylinder $(x,y,z,r)=(15,15,15,5)$ and n $(1,1,1)$ mm
Type 1	[-2.2,2,2]	1000 pts	900 pts	625 pts
Type 2	[-5,5]	1000 pts	900 pts	625 pts
Normal	standard deviation [μm]			
Type 1	1.1	1000 pts	900 pts	625 pts
Type 2	2.5	1000 pts	900 pts	625 pts

The second implementation and comparison is applied to points obtained from a series of real measurements by means of tactile coordinate measuring machine (CMM). Kawalec and Magdziag [22] reported comparison of methods to solve circle fitting problem by measuring a ring gauge as case study. There are two selected case studies for this real measurement approach: a reference ceramic sphere and an industrial cylinder. Figure 15 shows the measurement of the sphere and cylinder by tactile CMM.

For the ceramic sphere case in figure 15a, a calibrated ceramic sphere of a traceable tactile “ZEISS PRISMO” CMM with $MPE_E=2 \mu\text{m}+L/300 \mu\text{m}$ for stylus qualification was used with calibrated radius of 14.991 mm. The sphere case study is a good example of a common artifact for CMMs. The point cloud was obtained by tactile scanning. The point clouds to be fit are half-geometries, constituted from low density and high density points. The low density point cloud contains 312 points and the high density one contains total of 3435 points. The initial parameter of the fitting, for both LM and Chaos-LM method, has been chosen near the calibrated value. Initial x and y location are from the average of respectively x- and y-position of the points, and z location is selected from maximum z-position of the points minus the known nominal radius of 15 mm. For the cylinder case study (figure 15b), it is made of hardened steel having a nominal diameter of 6 mm. Total points obtained for the measurement of industrial cylinder were 190 points by circular path scanning strategy of three segments, including the equator line.

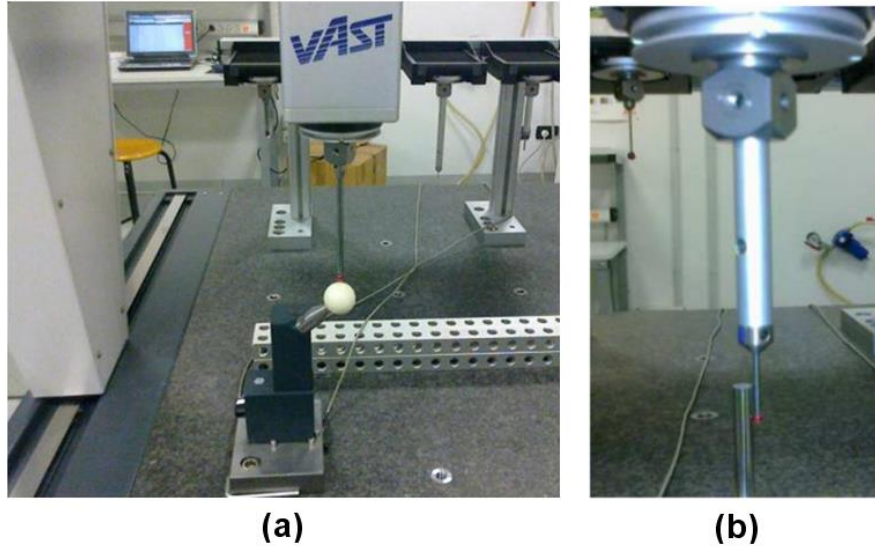


Figure 15: (a) sphere measurement and (b) industrial cylinder measurement.

The algorithm is implemented in MATLAB and run on an Intel Centrino Core 2 Duo 2.2 GHz. Center of circle and sphere is initialized by selecting the centroid of the cloud of points as initial guess. Moreover, the centroid is also the initial guess of a point on the axis of the cylinder. Initial estimation for radius is:

$$r_0 = \frac{1}{2} \left(\frac{(\max x_i - \min x_i) + (\max y_i - \min y_i)}{2} \right) \text{ for the circle;} \quad (8)$$

$$r_0 = \frac{1}{2} \left(\frac{(\max x_i - \min x_i) + (\max y_i - \min y_i) + (\max z_i - \min z_i)}{3} \right) \text{ for the sphere and cylinder} \quad (9)$$

Where r_0 is initial radius estimation, x_o, y_o, z_o are spatial coordinates of all the points. Only in the case of the cylinder, the initial guess for cosine direction of the axis is derived by fitting a 3D line to the point clouds as proposed by the NIST [5].

5.1 Performance improvement

Details of the performance improvement can be found in the previous paper [8]. A brief review of the results will be presented. In the simulation approach 100 cloud of points were generated for each geometry and kind of noise generation and fitted. From the result, Chaos-LM algorithm gives better fitting performance. The reason is that Chaos initialization encourages the initial guess of the solution to move to a better starting point. Thanks to the property of the chaotic motion which non-repeatedly searches through a set of states in a certain bounded domain [29], the search can escape from local optima. Sensitiveness of the final solution of LM method to where the initial guess starts is related to the Taylor approximation in the Gauss-Newton method. This approximation method highly depends on the non-linearity degree of the neighborhood. Since this approximation is usually applied up to the first term, the validity of this Taylor

approximation decreases for higher non-linear function. Because of this, a “trapped” condition during the search process can occur.

To visually illustrate the improved performance of the fitting by Chaos-LM algorithm, figure 16 shows the fitting results applied to the circle. From this figure, one can observe that the Chaos-LM fitting (figure 16b) finally lies in the middle of the point cloud. This is coherent with the fundamental behavior of least-square fitting which is an average over the considered data (in this case the point cloud). For the computation time, in general there are no significant differences between LM and Chaos-LM algorithm, except for the case of cylinder fitting results, where the computation time slightly increases compared to the LM method. Indeed, the sum of squared residuals is significantly reduced.

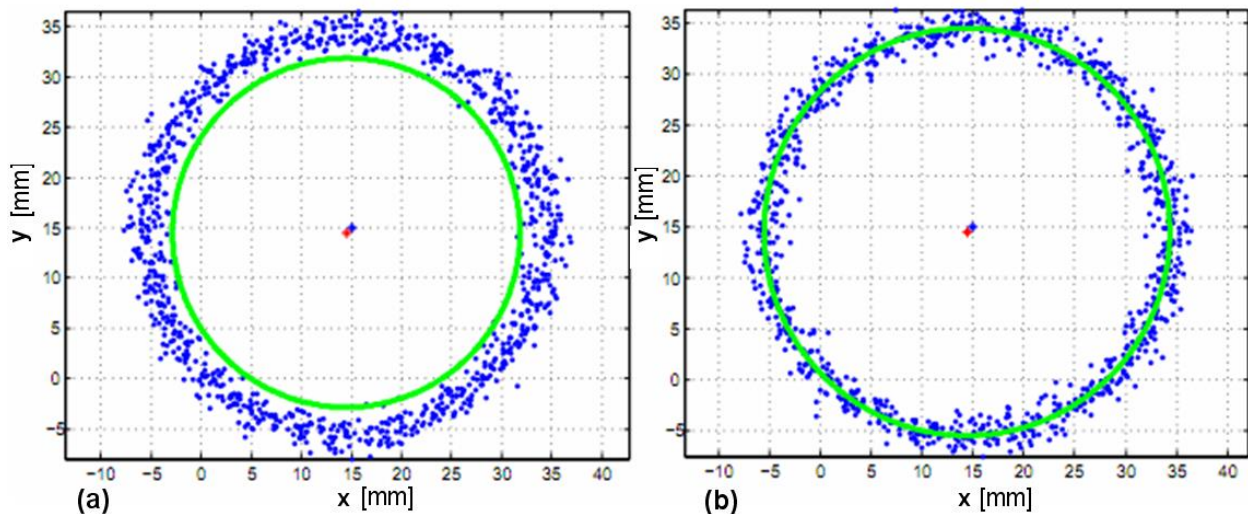


Figure 16: Substitute full circle fitting results of (a) LM method and (b) Chaos-LM [8].

For the half-geometry cases, a more difficult situation is observed. This is, in part, due to the not-so-good initial guess of the solution: in the case of full geometries, the initial estimation takes advantage of symmetry properties of the geometry. From the results [8], one can observe that the accuracy of fitting half-geometry point clouds is significantly improved by Chaos-LM method compared to LM method. Figure 17 graphically shows this improvement in the case of circle fitting. From the computation time point of view, Chaos-LM method has significantly higher CPU time compared to the LM case, hence the comparison goes further with the a convergence analysis.

Convergence curve analyses were carried out both for full- and half- geometries fitting. It is worth noting that the curves cannot reach the zero since there are errors on the points due to the simulated noise. Noise for the simulation of this analysis is uniformly distributed in the range between $-5\mu\text{m}$ and $5\mu\text{m}$. Some convergence graphs are re-presented in figure 18 and 19 for full and half-circle cases, respectively. Please note that, in the convergence graphs, the abscissa sums the iterations required by both Chaos initialization and the LM algorithm in the case of Chaos-LM method. For LM method, it corresponds to LM number of iterations, which ranges from 100 to 1000 iterations. For Chaos-LM, this axis corresponds to the number of chaos iteration in the range from 10 to 100 iterations. A clear “trapped” phenomenon in a local optima region can be

observed in both the case of full- and half-geometry fitting. On the other hand, Chaos-LM algorithm can escape from local optima as the number of iteration increases. The chaotic movement during the searching process can explore a wider region to find a better solution. Chaos-LM can show a significant improvement and convergence result without a large number of iteration increments. From the investigation of the convergence, the optimal number of chaos iterations is around 30.

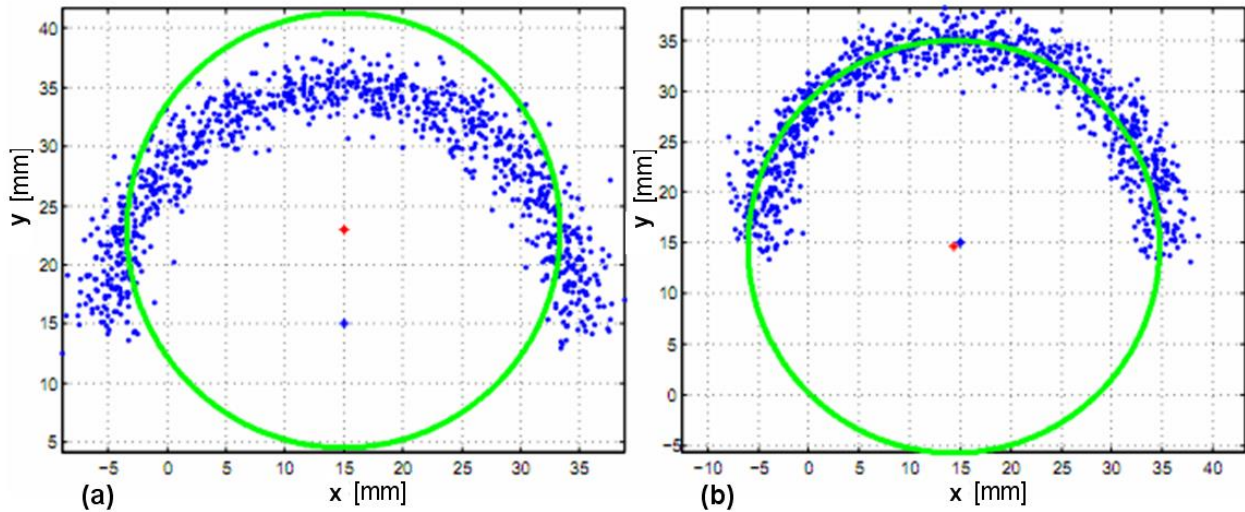


Figure 17: Substitute Half circle point cloud fitting results of (a) LM method and (b) Chaos-LM [8].

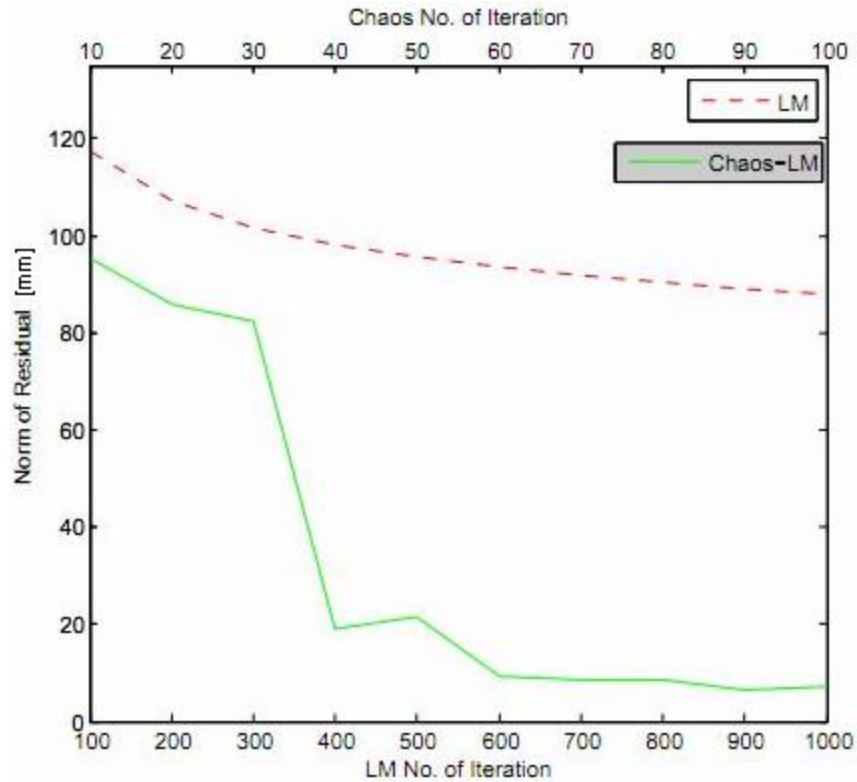


Figure 18: Convergence rate for fitting **full** circle point clouds [8].

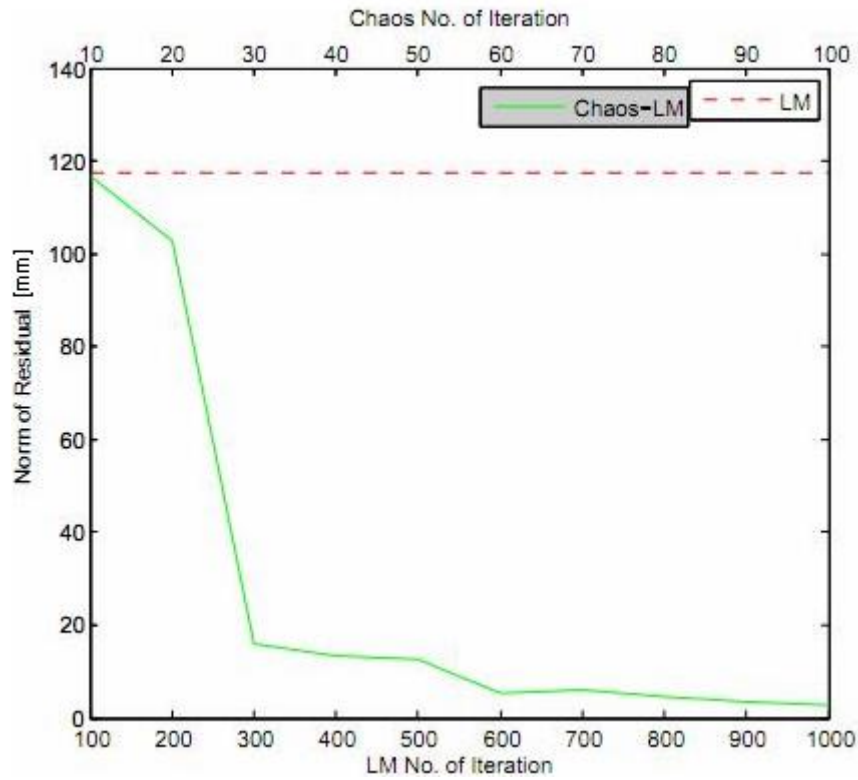


Figure 19: Convergence rate for fitting **half** circle point clouds [8].

From the real case study. Fitting improvements are also obtained by Chaos-LM method compared to the LM one [8]. Visualization of the fitting results of the case studies for the sphere and cylinder measurement is plotted in figure 20. In this figure, blue points are points from tactile scanning, red points are the fitted points, which is the fitting geometry, green line is estimated cylinder axis line.

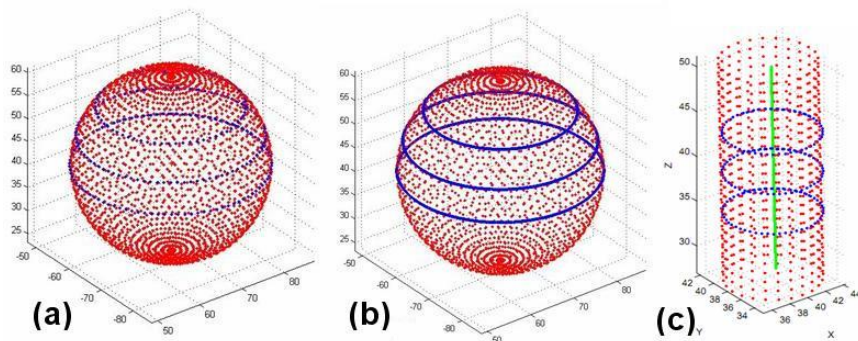


Figure 20: Fitting of (a) low-density sphere, (b) high-density sphere and (c) cylinder case.

Concluding, the CO is effective in improving the initial guess in non-linear LS fitting problem. Thanks to the chaotic dynamic that non-repeatedly moves through a set of trajectory states in a certain bounded domain [29], the CO has a higher chance to make the initial guess of the solution closer to the global optimum region. This is a unique property which is different compared to the common improvement heuristic search such as genetic algorithm, tabu search,

etc. [30]. They need larger number of iterations to increase the “visited feasible solution” inside the search space by which the computational time increases.

5.2 Performance evaluation of one-dimensional map functions

In this section, comparison of Chaos-LM algorithm performance with respect to different one-dimensional functions as chaotic motion generator of CO algorithm will be presented. The simulation data used for the comparison study is perturbed by noise applied to the nominal points. This noise is sampled from a uniform distribution in the range between $-5 \mu\text{m}$ and $5 \mu\text{m}$. This assumption includes the MPE (maximum permissible error) of a hypothetical CMM and the hypothetical part feature deviation. The complete results of the comparison, including both the norm of the residuals $\|\mathbf{r}\|$ and the CPU time, are presented in table 4. To compare the influence of different one-dimensional maps, an analysis of the variance (ANOVA) was carried out. Three replications (each of replica is mean value from 100 simulation runs) were carried out for each test. The treatments of the test are the continuous one-dimensional map. Identical to the previous section, response variables of the test is that the norm of the residual $\|\mathbf{r}\|$ and the CPU time. The test demonstrates that the map chosen significantly affect both $\|\mathbf{r}\|$ and CPU time, with the only exception of the CPU time for sphere fitting (figure 22b).

The comparisons for full-geometries fitting are presented as bar chart in figure 21, 22, and 23, and for half-geometries in figure 24, 25, and 26. From these figures, one can observe that, in general, the logistic-map is the most effective to solve the optimization problem in terms of $\|\mathbf{r}\|$, with some compromises of having higher computational time in certain cases. For example, in the case of half-circle fitting, the logistic map provides the best solution, but the computation time is significantly larger. In full-circle fitting (figure 21), Liebovitch and Intermittency chaos generation give the worst solution while for full-cylinder fitting (figure 23), Bernoulli Shift and Sine chaos generator give the worst solution. A solution in general similar is observed in the case of full-sphere fitting (figure 22). Higher variation in results is observed in the case of half-geometries compared to the full geometries. Liebovitch and Intermittency generator give the worst results for half-sphere and half-cylinder. Bernoulli Shift, Sine, and Circle generator gives the worst results of half-circle fitting. Table 4 gives the detail results of the comparison runs.

Table 4: Complete result of several types of one-dimensional map.

Random Error Type (μm)	Type of One-Dimensional Map	Full-Geometries Chaos and Levenberg-Marquardt Algorithm					
		Circle		Sphere		Cylinder	
		$\ \mathbf{r}\ (\mu\pm 3\sigma)$	CPU time ($\mu\pm 3\sigma$)	$\ \mathbf{r}\ (\mu\pm 3\sigma)$	CPU time ($\mu\pm 3\sigma$)	$\ \mathbf{r}\ (\mu\pm 3\sigma)$	CPU time ($\mu\pm 3\sigma$)
U[-5,5]	Logistic	3.3562±4.4067	0.5551±0.0308	6.2199±0.4602	0.5628±0.1088	5.7994±2.9641	0.6706±0.0515
	Tent	3.8789±1.3553	0.5400±0.0282	6.2625±0.5551	0.4987±0.0856	5.0455±1.9903	0.7874±0.1155
	Bernoulli Shift	28.4587±10.1690	0.4947±0.0941	7.5839±1.5888	0.6106±0.3284	88.752±3.2902	0.7665±0.0997
	Liebovitch	92.7066±6.2385	0.4787±0.0387	7.1002±0.5687	0.5377±0.0265	41.2997±10.5511	0.8015±0.1005

	Intermittency	92.9567±5.6253	0.4994±0.1139	7.1161±0.8001	0.4675±0.0266	3.3571±0.4035	0.7795±0.0443
	Chebyshev	37.4017±10.6941	0.5637±0.0794	6.975±0.7605	0.5527±0.0247	3.2422±0.4954	0.7885±0.0703
	Gaussian	3.8776±1.5751	0.5948±0.1065	6.3682±0.8676	0.5628±0.1068	2.9529±0.0444	0.7823±0.0527
	Sine	2.8379±1.0583	0.4947±0.0896	6.6963±1.2561	0.5531±0.0295	81.9015±11.0938	0.8223±0.0991
	ICMIC	27.3192±10.6311	0.4862±0.0732	6.9606±0.8579	0.5782±0.1159	2.9676±1.1675	0.8172±0.0383
	Circle	22.3999±10.0398	0.5216±0.0947	7.4993±1.7130	0.5840±0.1275	88.5412±4.1864	0.785±0.0915
Half-geometries Chaos and Levenberg-Marquardt Algorithm							
Random Error Type (um)	Type of One-Dimensional Map	Circle		Sphere		Cylinder	
		$\ r\ (\mu\pm 3\sigma)$	CPU time ($\mu\pm 3\sigma$)	$\ r\ (\mu\pm 3\sigma)$	CPU time ($\mu\pm 3\sigma$)	$\ r\ (\mu\pm 3\sigma)$	CPU time ($\mu\pm 3\sigma$)
U[-5,5]	Logistic	15.9923±12.0558	1.2212±0.1456	9.2163±6.3610	0.413±0.0247	18.7634±3.0584	1.2185±0.1001
	Tent	99.9067±10.5034	0.4945±0.0642	10.8683±2.3928	0.458±0.0601	18.8646±1.6585	1.6603±0.2450
	Bernoulli Shift	116.6437±6.0842	0.5704±0.1080	20.1410±10.4110	0.4742±0.0704	39.7236±3.7368	1.7059±0.2628
	Liebovitch	100.4388±12.9591	0.4843±0.0373	43.1376±3.7162	0.4667±0.0795	18.8829±1.5393	1.6815±0.1123
	Intermittency	48.8010±6.9925	0.5731±0.0601	43.1131±4.7619	0.5632±0.0349	18.9943±1.7687	1.6443±0.1796
	Chebyshev	22.7672±10.8399	0.5440±0.0267	27.5289±15.9982	0.5622±0.0382	18.8082±2.0556	1.6513±0.1084
	Gaussian	60.9419±10.9246	0.5017±0.1493	11.5014±10.2587	0.5802±0.0749	18.9913±1.2906	1.5953±0.2921
	Sine	115.0962±9.9298	0.4721±0.0462	20.0862±10.0986	0.5693±0.0880	28.5602±12.6451	1.5441±0.1390
	ICMIC	15.2252±5.8807	0.5587±0.0316	28.6431±14.0605	0.4731±0.0362	19.142±2.5265	1.5227±0.3441
	Circle	116.5382±6.7659	0.4803±0.0500	13.7452±12.5602	0.4661±0.0510	18.4772±4.1249	1.6762±0.2146

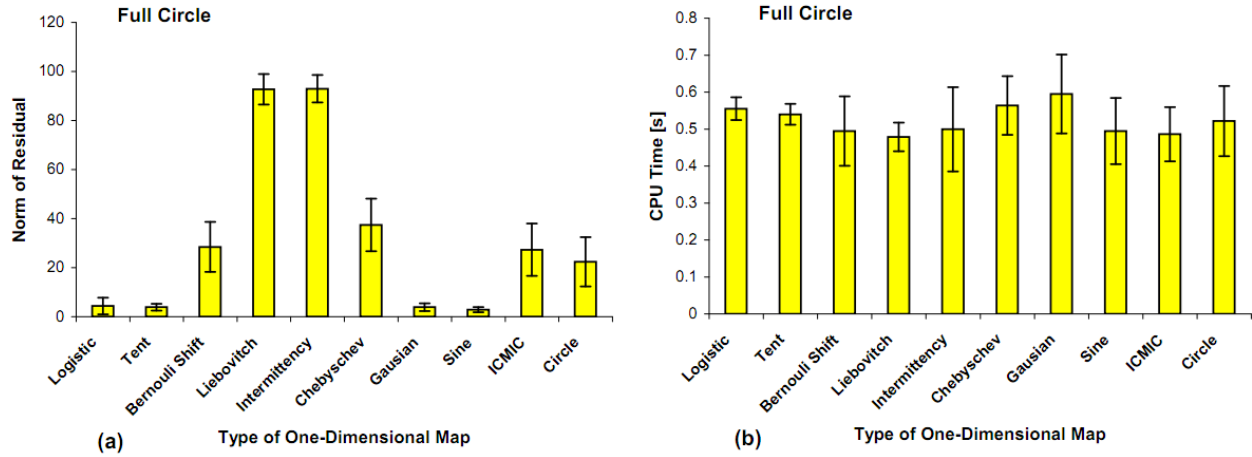


Figure 21: Comparison of type of one-dimensional map for Chaos-LM method for full circle point cloud fitting. (a) Norm of residual, (b) CPU time.

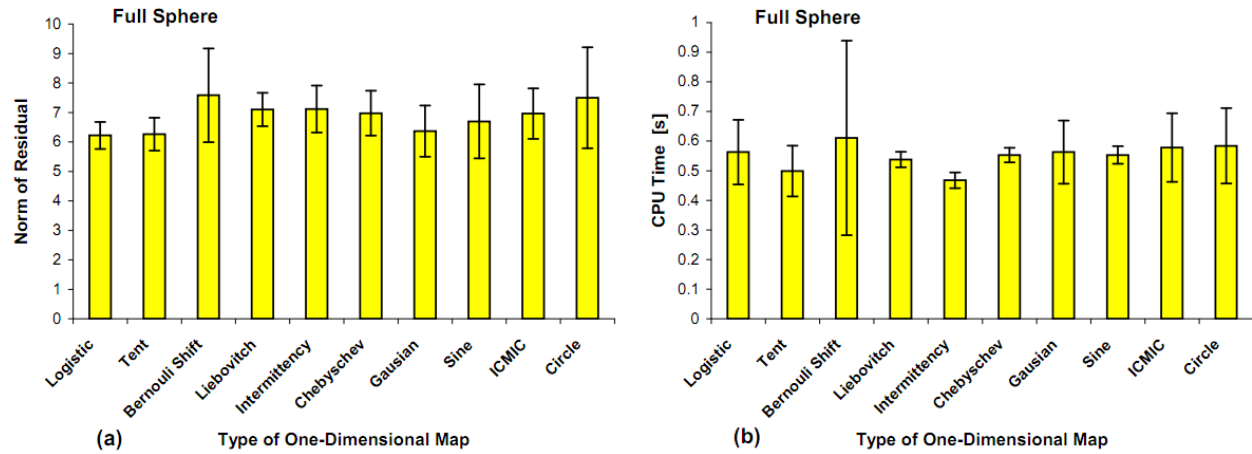


Figure 22: Comparison of type of one-dimensional map for Chaos-LM method for full sphere point cloud fitting. (a) Norm of residual, (b) CPU time.

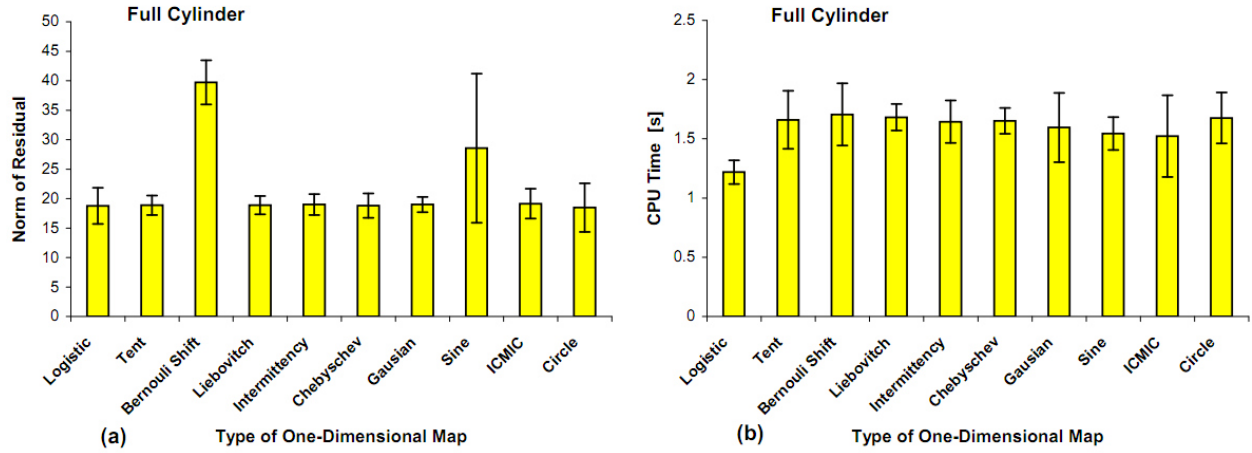


Figure 23: Comparison of type of one-dimensional map for Chaos-LM method for full cylinder point cloud fitting. (a) Norm of residual, (b) CPU time.

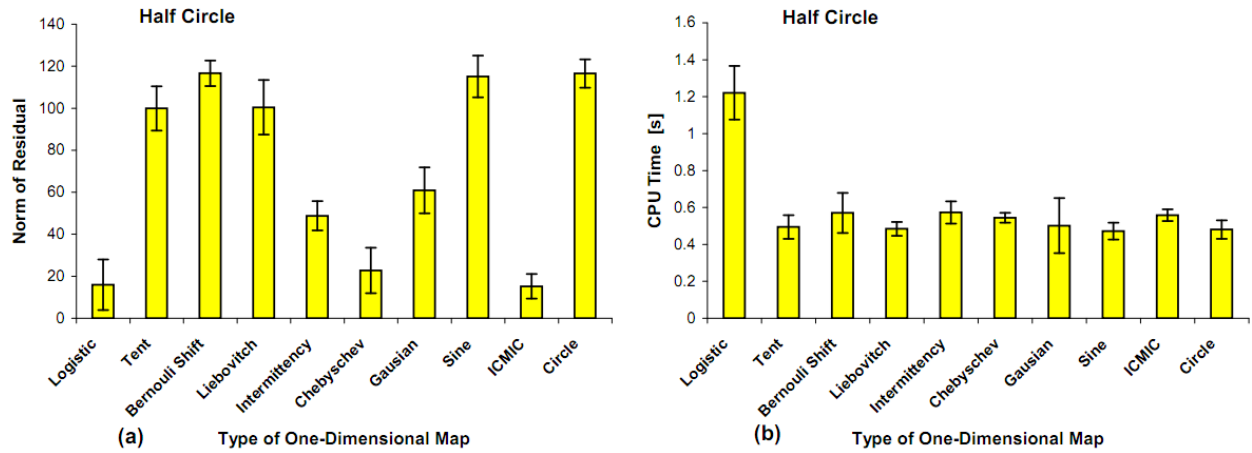


Figure 24: Comparison of type of one-dimensional map for Chaos-LM method for half circle point cloud fitting. (a) Norm of residual, (b) CPU time.

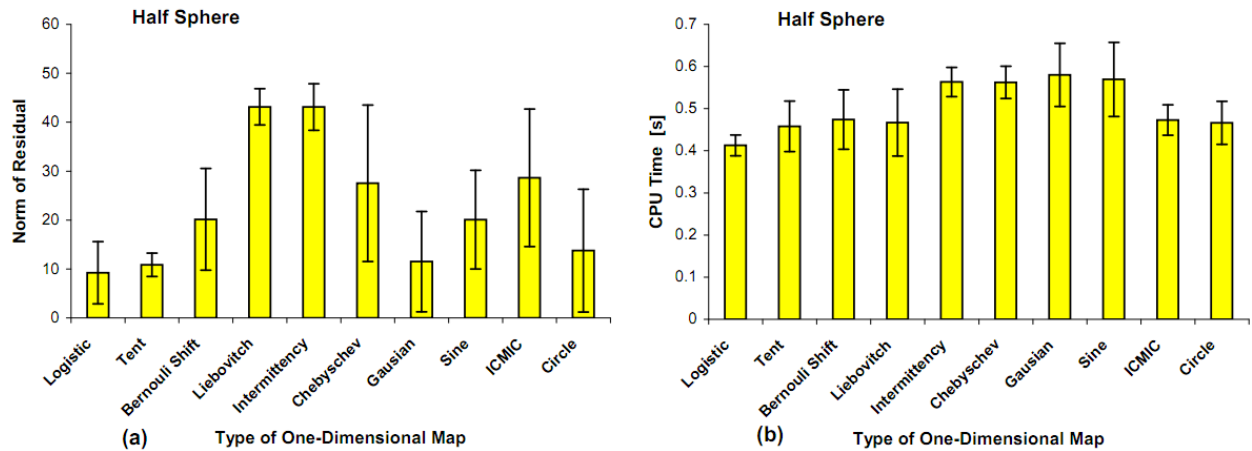


Figure 25: Comparison of type of one-dimensional map for Chaos-LM method for half sphere point cloud fitting. (a) Norm of residual, (b) CPU time.

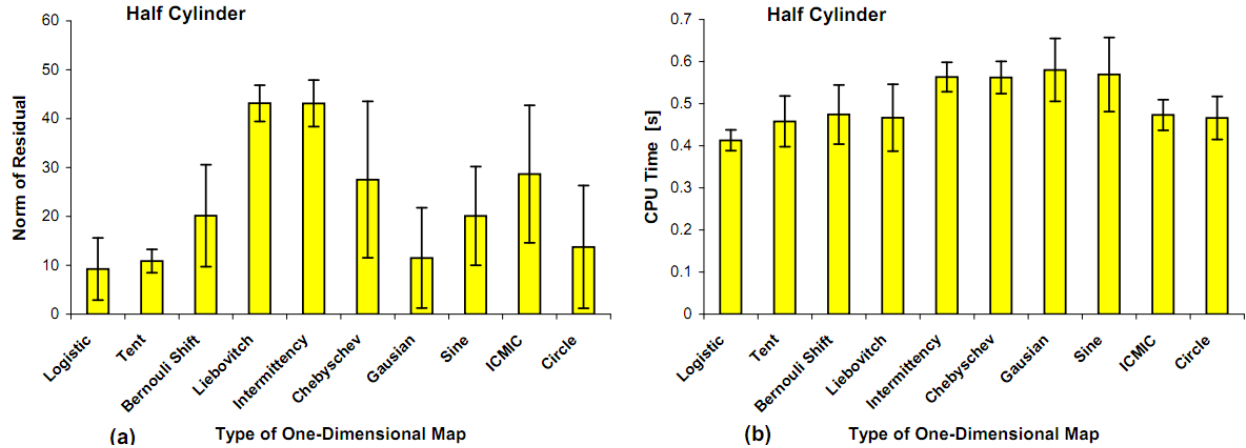


Figure 26: Comparison of type of one-dimensional map for Chaos-LM method for half cylinder point cloud fitting. (a) Norm of residual, (b) CPU time.

In addition, the performance comparison among one-dimensional map of CO method is also carried out on real measurement data. The results are shown in figure 27, 28, and 29 for real case studies of sphere measurement with low-density points, with high-density points and industrial cylinder measurement, respectively. Table 3 presents the detailed results of the comparison study. From these graphs, one can observe that the comparisons of the one-dimensional map performance are similar to those proposed earlier. Logistic map, Liebovitch map and Gaussian map obtained the best result for the norm of residual in the case of both the sphere fitting from low and high-density points. Instead, in the case of industrial cylinder fitting, Tent, Intermittency and Chebyshev map give the least norm of residual. Logistic map does not perform well in this case. This could be caused by small number of points to fit compared to other case studies. About the CPU time needed to fit the data, there is no significant difference among the different one-dimensional map functions.

Table 3: Detail results of comparison of one-dimensional map for real measurement case study.

Random Error Type	Type of One-Dimensional Map	Chaos and Levenberg-Marquardt Algorithm					
		Sphere Low Density		Sphere High Density		Industrial Cylinder	
		$\ r\ (\mu \pm 3\sigma)$	CPU time ($\mu \pm 3\sigma$)	$\ r\ (\mu \pm 3\sigma)$	CPU time ($\mu \pm 3\sigma$)	$\ r\ (\mu \pm 3\sigma)$	CPU time ($\mu \pm 3\sigma$)
U [-0.005, 0.005]	Logistic	0.5139 ± 0.5044	0.4991 ± 1.961	2.2776 ± 1.882	3.8707 ± 0.33	2.1981 ± 0.4822	8.4489 ± 2.3731
	Tent	1.2014 ± 0.8465	0.4683 ± 0.0221	4.2608 ± 2.5093	4.0577 ± 0.025	0.4197 ± 0.7331	8.2408 ± 1.5865
	Bernoulli Shift	17.8537 ± 0.02	0.46823 ± 0.024	57.3112 ± 1.8702	4.0531 ± 0.037	3.6392 ± 0.0079	6.9392 ± 1.99
	Liebovitch	0.7122 ± 0.5861	0.4629 ± 0.0167	1.9955 ± 0.6037	4.0732 ± 0.052	3.3185 ± 1.6297	8.3943 ± 0.2994
	Intermittency	3.1645 ± 0.9661	0.4736 ± 0.0385	7.8977 ± 3.2939	4.358 ± 0.9656	0.3209 ± 0.6761	8.5420 ± 0.1524
	Chebyshev	1.7974 ± 1.8641	0.4916 ± 0.0634	4.542 ± 3.989	4.7134 ± 3.034	0.4340 ± 0.9100	8.2558 ± 1.504

Gaussian	0.5855 ± 0.4368	0.4687 ± 0.0222	1.7241 ± 2.0967	5.2739 ± 2.9522	3.6351 ± 0.02	8.514 ± 0.4319
Sine	17.5325 ± 0.6418	0.4642 ± 0.082	55.2934 ± 6.6626	4.6047 ± 2.7276	3.6392 ± 0.02	8.9919 ± 1.9286
ICMIC	17.73143 ± 0.1382	0.4676 ± 0.0282	2.0303 ± 0.5039	5.0282 ± 4.7533	0.455 ± 0.05	8.5241 ± 0.0868
Circle	17.8149 ± 0.1784	0.4629 ± 0.072	57.3171 ± 0.7026	4.1288 ± 0.127	3.6391 ± 0.02	9.0029 ± 1.313

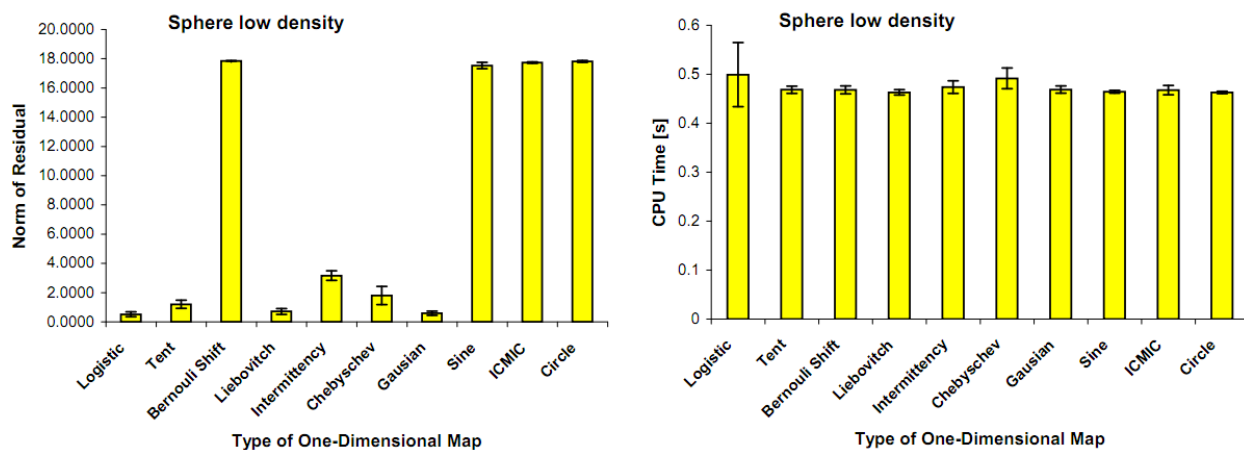


Figure 27: Comparison of type of Chaos one-dimensional map for sphere measurement with low density points (312 points). (a) Norm of residual, (b) CPU time.

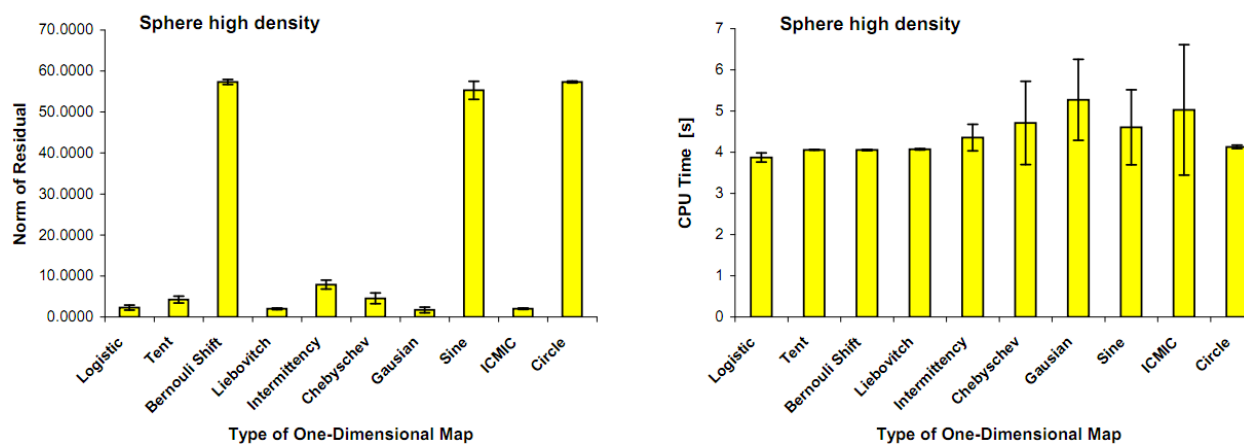


Figure 28: Comparison of type of Chaos one-dimensional map for sphere measurement with high density points (3435 points). (a) Norm of residual, (b) CPU time.

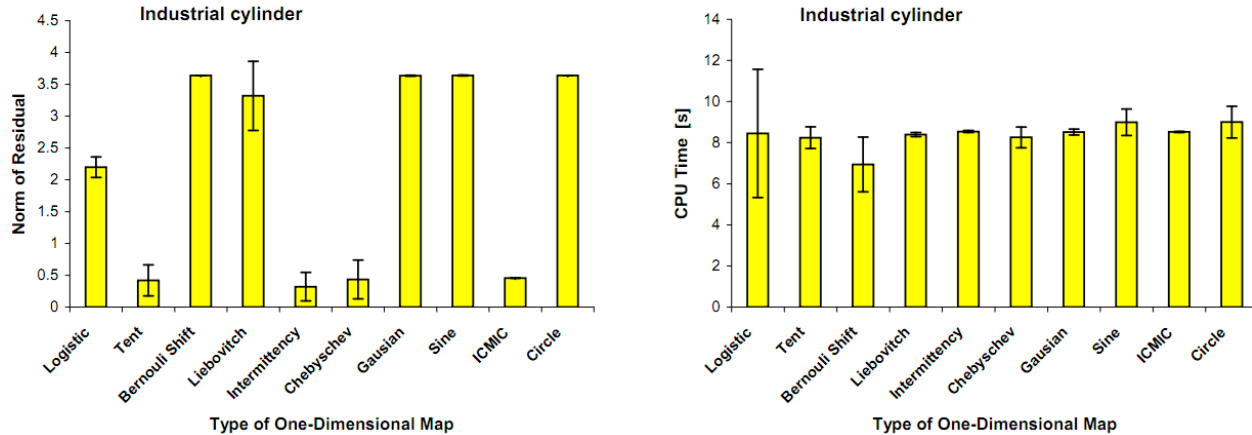


Figure 29: Comparison of type of Chaos one-dimensional map for industrial cylinder measurement (190 points). (a) Norm of residual, (b) CPU time.

6. Concluding Remarks

In this article, the problem of fitting non-linear geometries has been discussed. This fitting procedure is critical in the whole chain of dimensional metrology. The reason is that many product quality inspections are realized by means of dimensional quality inspection. Modern metrology instruments can capture high density clouds of points in short time. Hence, this fitting problem becomes more and more relevant. Three non-linear geometries are considered in this work, the circle, the sphere and the cylinder, due to their diffused use in applications such as metrological calibration and mechanical assembly. Both cases of fitting full- and half-geometries are addressed. It has been shown that a performance improvement of the fitting results from simulated and real measurement data can be obtained by combining chaos optimization and LM algorithm compared to a single LM algorithm. The LM method is in many cases trapped and early converged during the optimization process, in particular in the case of half-geometries fitting. In this situation, no improvement of the result can be obtained by increasing the number of iterations. Comparison among various type of one-dimensional map which determine the chaos motion is presented as well. A total of 10 one-dimensional map functions are considered. This comparison study points out that, in general, the logistic map function performs the best in almost all situations compare to others type of one-dimensional maps. The future direction of this work is the identification of the link between the non-linear problem and chaos properties such that an adaptive region bounding and chaotic motion generation can be better determined.

Acknowledgements

Financial support to this work has been provided as part of the project REMS - Rete Lombarda di Eccellenza per la Meccanica Strumentale e Laboratorio Esteso, funded by Lombardy Region (Italy), CUP: D81J10000220005 and AMala – Advanced Manufacturing Laboratory, funded by Politecnico di Milano (Italy), CUP: D46D13000540005.

References

- [1] Kunzmann H, Pfeifer T, Schmitt R, Schwenke H, Weckenmann A 2005 Productive Metrology-Adding Value to Manufacture *CIRP Ann. Manuf. Techn.* 54(2) 155 – 168.
- [2] Pereira P H 2012 Cartesian Coordinate Measuring Machines, in Hocken R. J., Pereira, P. H (Eds), *Coordinate Measuring Machines and Systems 2nd ed*, (Florida: CRC Press) 57-79.
- [3] Morse E 2012 Operating a Coordinate Measuring Machine, in Hocken R. J., Pereira, P. H (Eds), *Coordinate Measuring Machines and Systems 2nd ed*, (Florida: CRC Press) 81-91.
- [4] Wilhelm R G, Hocken R, Schwenke H 2001 Task Specific Uncertainty in Coordinate Measurement *CIRP Ann. Manuf. Techn.* 50(2) 553-563.
- [5] Shakarji C M 1998 Least-Squares Fitting Algorithms of the NIST Algorithm Testing System, *Journal of Research NIST* 103 633-641.
- [6] Hoppe H, DeRose T, Duchamp T, McDonald J, Stuetzle W 1992 Surface reconstruction from unorganized points *ACM* 26(2) 71-78.
- [7] Moroni G, Petrò S, Tolio T 2011 Early cost estimation for tolerance verification *CIRP Ann. Manuf. Techn.* 60(1) 195-198.
- [8] Moroni G, Syam W P, Petro S. 2014 Performance improvement for optimization of the non-linear geometric fitting problem in manufacturing metrology *Measurement Science and Technology* 25 085008.
- [9] ISO 10360-4:2010 2010 Acceptance and re-verification test for coordinate measuring machines: CMM used in scanning measuring mode.
- [10] ISO 10360-5:2010 2010 Acceptance and re-verification test for coordinate measuring machines: CMM using single and multiple styluses contacting probing system.
- [11] Moroni G, Petrò S, Syam W P. 2014 Four-axis micro measuring systems performance verification *CIRP Annals-Manufacturing Technology* Vol. 63 (1), 485-488.
- [12] Moroni G, Syam W P, Petrò S. 2015 On Performance Verification of Simultaneous 4-axis 3D Geometric Optical Instrument *LAMDAMAP XI, 17-18 March 2015, Huddersfield, UK*.
- [13] Whitney D E 2004 *Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development* (USA: Oxford University Press).
- [14] Marquardt D W 1963 An Algorithm for Least-Squares Estimation of Non-linear Parameters, *J. Soc. Ind. Appl. Math.* 11(2) 431-441.

- [15] Nash J C 1979 Compact Numerical Methods for Computers Linear Algebra and Function Minimization (Bristol: Adam Higler Ltd).
- [16] Rardin R L 2006 Optimization in Operation Research (New York: Addison-Wesley).
- [17] Syam W P and Al-Harkan I M 2010 Comparison of Three Meta Heuristics to Optimize Hybrid Flow Shop Scheduling Problem with Parallel Machines, *World Academy of Science, Engineering and Technology* 62 271-278.
- [18] Luo Y Z, Tang G J, Zhou N L 2008 Hybrid approach for solving systems of nonlinear equations using chaos optimization and quasi-Newton method *Applied Soft Computing* 8(2) 1068-1073.
- [19] Tavazoei M S and Haeri M 2007 An optimization algorithm based on chaotic behavior and fractal nature *Comput. Appl. Math.* 206(2) 1070-1081.
- [20] Yang D, Li G, Cheng G 2009 On the efficiency of chaos optimization algorithms for global optimization *Chaos, Solitons & Fractals* 34(4) 1366-1375.
- [21] E. Ott, Chaos in Dynamical Systems, Cambridge University Press, Cambridge, UK, 2002.
- [22] A. Erramilli, R.P. Singh, P. Pruthi, Modeling packet traffic with chaotic maps, Royal Institute of Technology, ISRN KTH/IT/R-94/18-SE, Stockholm-Kista, Sweden, August 1994.
- [23] D. He, C. He, L. Jiang, H. Zhu, G. Hu. 2001. Chaotic characteristic of a one-dimensional iterative map with infinite collapses, *IEEE Transactions on Circuits and Systems*, Vol. 48 (7) 1076-1085.
- [24] M. Bucolo, R. Caponetto, L. Fortuna, M. Frasca, A. Rizzo. 2002. Does chaos work better than noise?, *IEEE Circuits and Systems Magazine*, Vol. 2 (3) 4-19.
- [25] R.L. Devaney, An Introduction to Chaotic Dynamical Systems, Addison-Wesley, USA, 2003.
- [26] Liebovitch LS and Tooth TI. 1990. Ion Channel Kinetics: Random or Deterministic Process?. *Biophysical Journal*, Vol. 57, 317a.
- [27] Schuster, H. G. 2005. Deterministic Chaos: An Introduction, 2nd Ed. Wiley-VCH Verlag GmbH: Germany.
- [28] Kruth J P, Van Gestel N, Bleys P, Welkenhuyzen F. 2009 Uncertainty determination for CMMs by Monte Carlo simulation integrating feature form deviations *CIRP Ann. Manuf. Techn.* 58(1) 463-466.
- [29] Jiang B L W 1998 Optimizing complex functions by chaos search. *Cybern. Syst.* 29(4) 409-419.

[30] Trafalis T B and Kasap S 2002 A novel metaheuristic approach for continuous global optimization *J. Global. Optim.* 23 171-190.