



Searching the Optimal Parameters of a 3D Scanner Through Particle Swarm Optimization

João Braun^{1,2}(✉) , José Lima^{2,3} , Ana I. Pereira² , Cláudia Rocha³ ,
and Paulo Costa^{1,3} 

¹ Faculty of Engineering, University of Porto, Porto, Portugal
`paco@fe.up.pt`

² Research Centre in Digitalization and Intelligent Robotics,
Instituto Politécnico de Bragança, Bragança, Portugal
`{jbneto,jllima,apereira}@ipb.pt`

³ INESC-TEC - INESC Technology and Science, Porto, Portugal
`claudia.d.rocha@inesctec.pt`

Abstract. The recent growth in the use of 3D printers by independent users has contributed to a rise in interest in 3D scanners. Current 3D scanning solutions are commonly expensive due to the inherent complexity of the process. A previously proposed low-cost scanner disregarded uncertainties intrinsic to the system, associated with the measurements, such as angles and offsets. This work considers an approach to estimate these optimal values that minimize the error during the acquisition. The Particle Swarm Optimization algorithm was used to obtain the parameters to optimally fit the final point cloud to the surfaces. Three tests were performed where the Particle Swarm Optimization successfully converged to zero, generating the optimal parameters, validating the proposed methodology.

Keywords: Nonlinear optimization · Particle swarm optimization · 3D scan · IR sensor

1 Introduction

The demand for 3D scans of small objects has increased over the last few years due to the availability of 3D printers for regular users. However, the solutions are usually expensive due to the complexity of the process, especially for sporadic users. There are several approaches to 3D scanning, with the application, in general, dictating the scanning system's requirements. Thus, each of them can differ concerning different traits such as acquisition technology, the structure of the system, range of operation, cost, and accuracy. In a more general way, these systems can be classified as contact or non-contact, even though there are several sub-classifications inside these two categories, which the reader can find with more detail in [2].

Two common approaches for reflective-optical scanning systems are triangulation and time-of-flight (ToF). According to [6], the range and depth variation in triangulation systems are limited, however, they have greater precision. In contrast, ToF systems have a large range and depth variation at the cost of decreased precision. The triangulation approach works, basically, by projecting a light/laser beam over the object and capturing the reflected wave with a digital camera. After that, the distance between the object and the scanning system can be computed with trigonometry, as the distance between the camera and the scanning system is known [6]. On the other hand, the accuracy of ToF-based systems is mainly determined by the sensor's ability to precisely measure the round-trip time of a pulse of light. In other words, by emitting a pulse of light and measuring the time that the reflected light takes to reach the sensor's detector, the distance between the sensor and the object is measured. Besides, still regarding ToF-based systems, there is another approach to measure the distance between the sensor and the object based on the Phase-Shift Method, which essentially compares the phase shift between the emitted and reflected electromagnetic waves.

Each approach has its trade-offs, which a common one is between speed and accuracy in these types of scanning systems. By increasing the scanning speed of the system, the accuracy will decrease, and vice-versa. This is mitigated by having expensive rangefinders with higher sampling frequencies. In addition, accuracy during acquisition can be heavily affected by light interference, noise, and the angle of incidence of the projected beam in the object being much oblique. Therefore, a controlled environment, and a high-quality sensor and circuit must be used to perform quality scans. However, the angle of incidence is less controllable since it depends on the scanning system and the object (trade-off of the system's structure). In general, these types of 3D scanning systems are very expensive, especially when even the cheaper ones are considered costly by the regular user. A particular example includes a low-cost imagery-based 3D scanner for big objects [18].

A low-cost 3D scanning system that uses a triangulation approach with an infra-red distance sensor targeting small objects, therefore having limited range and requiring large accuracy, was proposed and validated both in simulation [5] and real scenarios, where the advantages and disadvantages of this architecture were also addressed. Although the proposed system already has good accuracy, it is possible to make some improvements with a multivariable optimization approach. This is possible because the system makes some assumptions to perform the scan that, although are true in simulation, do not necessarily hold in a real scenario. One example is the object not being exactly centered on the scanning platform. The assumptions, as well as the problem definition, are described in Subsect. 4.1. The reasons why PSO was chosen instead of other approaches are addressed in Subsect. 4.2.

The paper is structured as follows: a brief state of the art is presented in Sect. 2. Section 3 describes the 3D scanning system and how it works. The optimization problem is defined, and a solution is proposed in Sect. 4. The results are described in Sect. 5. Finally, the last section presents the conclusion and future works.

2 State of the Art

As the field of optimization is vast, there are works in several fields of knowledge. Therefore, in this section, the objective is to give a brief literature review on multivariable optimization. Thus, some recent examples follow below.

Regarding optimization for shape fitting, [9] proposed a 3D finite-element (FE) modeling approach to obtain models that represent the shape of grains adaptively. According to the authors, the results show that the proposed approach supports the adaptive FE modeling for aggregates with controllable accuracy, quality, and quantity, which indicates reliable and efficient results. Moreover, authors in [11], proposed an approach that combined an optimization-based method with a regression-based method to get the advantages of both approaches. According to the authors, model-based human pose estimation is solved by either of these methods, where the first leads to accurate results but is slow and sensitive to parameter initialization, and the second is fast but not as accurate as of the first, requiring huge amounts of supervision. Therefore, they demonstrated the effectiveness of their approach in different settings in comparison to other state-of-the-art model-based pose human estimation approaches. Also, a global optimization framework for 3D shape reconstruction from sparse noisy 3D measurements, that usually happens in range scanning, sparse feature-based stereo, and shape-from-X, was proposed in [13]. Their results showed that their approach is robust to noise, outliers, missing parts, and varying sampling density for sparse or incomplete 3D data from laser scanning and passive multi-view stereo.

In addition, there are several works in multivariable optimization for mechanical systems. For instance, the characteristics, static and dynamic, of hydrostatic bearings were improved with controllers that were designed for the system with the help of a multivariable optimization approach for parameter tuning [16]. The authors proposed a control approach that has two inputs, a PID (Proportional Integral Derivative) sliding mode control and feed-forward control, where the parameters of the first input, were optimized with Particle Swarm Optimization (PSO). Their results performed better compared with results from the literature which had PID control. In addition, a system that consists of multi-objective particle swarm optimized neural networks (MOPSONNS) was proposed in [8] to compute the optimal cutting conditions of 7075 aluminum alloys. The system uses multi-objective PSO and neural networks to determine the machining variables. The authors concluded that MOPSONNS is an excellent optimization system for metal cutting operations where it can be used for other difficult-to-machine materials. The authors in [20] proposed a multi-variable Extremum Seeking Control as a model-free real-time optimizing control approach for operation of cascade air source heat pump. They used for the scenarios of power minimization and coefficient of performance maximization. Their approach was validated through simulation which their results showed good convergence performance for both steady-state and transient characteristics. The authors in [7] applied two algorithms, continuous ant colony algorithm (CACA) and multivariable regression, to calculate optimized values in back analysis of several

geomechanical parameters in an underground hydroelectric power plant cavern. The authors found that the CACA algorithm showed better performance. To improve the thermal efficiency of a parabolic trough collector, authors in [1] presented a multivariate inverse artificial neural network (ANNim) to determine several optimal input variables. The objective function was solved using two meta-heuristic algorithms which are Genetic Algorithm (GA) and PSO. The authors found that although ANNim-GA runs faster, ANNim-PSO has better precision. The authors stated that the two presented models have the potential to be used in intelligent systems.

Finally, there are optimization approaches in other areas such as control theory, chemistry, and telecommunication areas. Here are some examples. In control theory, by employing online optimization, a filtering adaptive tracking control for multivariable nonlinear systems subject to constraints was proposed in [15]. The proposed control system maintains tracking mode if the constraints are not violated, if not, it switches to constraint violation avoidance control mode by solving an online constrained optimization problem. The author showed that the switching logic avoids the constraints violation at the cost of tracking performance. In chemistry, artificial neural networks were trained under 5 different learning algorithms to evaluate the influence of several input variables to predict the specific surface area of synthesized nanocrystalline NiO-ICG composites, where the model trained with the Levenberg-Marquardt algorithm was chosen as the optimum algorithm for having the lowest root means square error [17]. It was found that the most influential variable was the calcination temperature. At last, in telecommunications, Circular Antenna Array synthesis with novel metaheuristic methods, namely Social Group Optimization Algorithm (SGOA) and Modified SGOA (MSGOA), was presented in [19]. The authors performed a comparative study to assess the performance of several synthesis techniques, where SGOA and MGSOA had good results.

As there aren't any related works similar to our approach, the contribution of this work is to propose a procedure to find the optimal parameters (deviations) of a 3D scanner system through PSO, guaranteeing a good fit for point clouds over their respective scanned objects.

3 3D Scanning System

The scanning system works basically in two steps. First, the object is scanned and the data, in spherical coordinates, is saved in a point cloud file. After this, a software program performs the necessary transformations to cartesian coordinates and generates STL files that correspond to the point clouds. The flowchart of the overall STL file generation process is presented in Fig. 1a. A prototype of the low-cost 3D scanner, illustrated in Fig. 1b, was developed to validate the simulated concept in small objects [5].

The architecture regarding the components and the respective communication protocols of the 3D scanning system can be seen in Fig. 2.

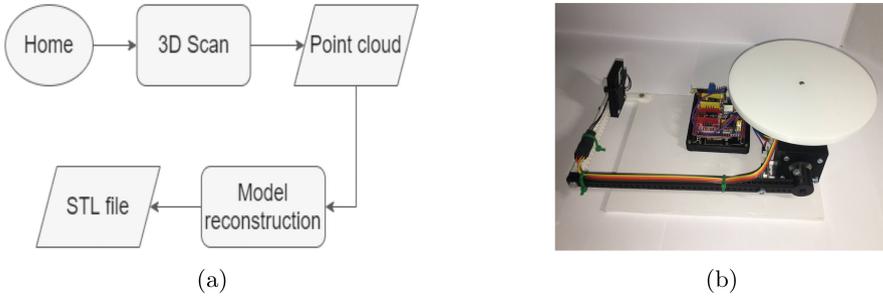


Fig. 1. Generation of an object’s STL file: (a) flowchart of the overall process, and (b) prototype of the low cost 3D scanner.

The system is composed of a rotating structure supporting the object that is going to be scanned and another articulated structure supporting the sensor that performs the data acquisition. The rotating structure is actuated by a stepper motor that precisely rotates it continuously until the end of one cycle, corresponding to 360° . An articulated part whose rotating axis is aligned with the center of the rotating structure, having attached an optical sensor that measures the distance to the object, is also actuated with a stepper motor which moves in steps of 2° (configurable). The proposed optical sensor was selected since it is a combination of a complementary metal-oxide-semiconductor image sensor and an infrared light-emitting diode (IR-LED) that allow a reduced influence by the environment temperature and by the object’s reflectivity. Both stepper motors used on this approach are the well-known NEMA17, controlled by the DRV8825 driver. Moreover, a limit switch sensor was added to be a reference to the position of 0° . Over a cycle, the sensor obtains 129 measurements, each one taking 50 ms. The rotating plate’s speed is approximately $55^\circ/s$ (configurable). The system is controlled by a low-cost ESP32 microcontroller, transferring to the PC, by serial or Wi-Fi, the measured distance to the object, the angle between the sensor and the rotating structure’s angle.

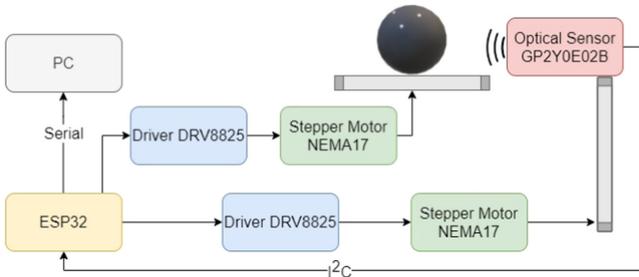


Fig. 2. Diagram illustrating the components of the system and the respective communication protocols used.

The dynamics of the 3D scanner is illustrated throughout a flowchart, which can be seen in Fig. 3. The rotating structure keeps rotating until it finishes one cycle of 360° . After this, the system increments the sensor's angle by 2° . This dynamic keeps repeating until the sensor's angle reaches 90° . Afterward, the data, in spherical coordinates, is saved to a point cloud file. The data conversion to Cartesian coordinates is explained in detail in Sect. 4. The algorithm that generates the STL files is outside of the scope of this work, for further information the reader is referred to [5].

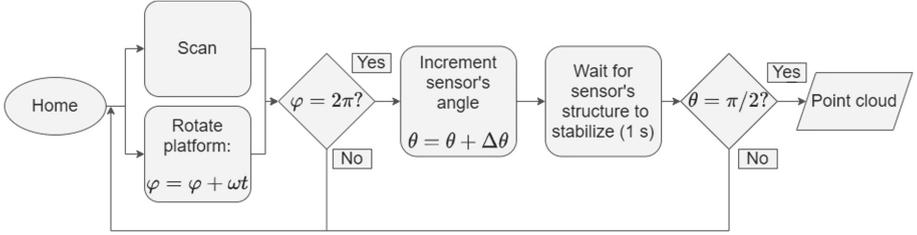


Fig. 3. 3D scanner dynamics flowchart.

4 Methodology

The notation used in this work to explain the system considers spherical coordinates, (l, θ, φ) . The scanning process starts by placing the object on the center of the rotating plate ($P = (0, 0, h_p)$), which is directly above the global reference frame. The system is represented in Fig. 4, from an XZ plane view. First, the slope's angle measured from the X axis to the Z axis, with range $[0, \pi/2]$ rad, is θ . Also, φ is the signed azimuthal angle measured from the X axis, in the XY plane, with range $[0, 2\pi]$. The parameter h_p represents the distance from the rotating plate to the arm rotation point, and h_s is the perpendicular distance from the arm to the sensor. The parameter l defines the length of the mechanical arm and d represents the distance from the sensor to the boundary of the object which is located on the rotating plate.

The position of the sensor s (red circle of Fig. 4) $X_s = (x_s, y_s, z_s)$, in terms of the parameters (l, θ, φ) , can be defined as follows, assuming the reference origin at the center of the rotating base:

$$\begin{cases} x_s = (l \cos(\theta) - h_s \sin(\theta)) \cos(\varphi) \\ y_s = (l \cos(\theta) - h_s \sin(\theta)) \sin(\varphi) \\ z_s = l \sin(\theta) + h_s \cos(\theta) - h_p \end{cases} \quad (1)$$

Therefore, $X_p = (x_p, y_p, z_p)$, which is the position of a scanned point in a given time in relation to the rotating plate reference frame, is represented by a translation considering $\|X_s\|$ and d . Thus, it is possible to calculate X_p as:

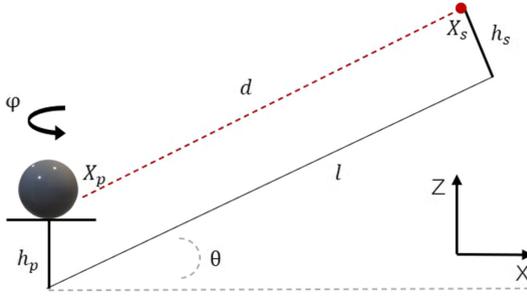


Fig. 4. Mechanical design of the proposed system. XZ plane view. Red circle represents the sensor and the dotted line represents its beam.

$$X_p = \left(1 - \frac{d}{\|X_s\|}\right) X_s \tag{2}$$

where $\|X_s\| = \sqrt{x_s^2 + y_s^2 + z_s^2}$, which represents the euclidean norm of (x_s, y_s, z_s) .

4.1 Problem Definition

The described calculations to obtain the position of a given scanned point X_p made several assumptions that, although always true in a simulated scenario, do not necessarily hold in real environments. If these assumptions would not be met, the coordinate transformations would not hold, and, therefore the point cloud and its mesh would become distorted. These assumptions, and consequently the optimization problem, are going to be described in this subsection.

First, there may exist an offset in θ (arm’s rotation angle), denoted by θ_{offset} , because the resting position of the arm can be displaced from the initial intended position. Following the same reasoning, an offset in φ , represented by φ_{offset} , is also possible, as the rotating plate can be displaced at the beginning of the scanning procedure. Finally, the scanned object may be displaced from the center of the rotating plate. This offset can be taken into account as an offset in the x and y coordinates constituting the sensor’s position, x_{offset} and y_{offset} respectively, as the effect would be the same as applying an offset to the object’s position. Thus, Eq. (1) is updated to Eq. (3) according with these offsets:

$$\begin{cases} x_s = (l \cos(\theta') - h_s \sin(\theta')) \cos(\varphi') + x_{\text{offset}} \\ y_s = (l \cos(\theta') - h_s \sin(\theta')) \sin(\varphi') + y_{\text{offset}} \\ z_s = l \sin(\theta') + h_s \cos(\theta') - h_p \end{cases} \tag{3}$$

where $\theta' = \theta + \theta_{\text{offset}}$ and $\varphi' = \varphi + \varphi_{\text{offset}}$.

Finally, in theory, the sensor must be orthogonal to the rotating arm. However, the sensor may become rotated γ_{offset} degrees during the assembly of the real system. Thus, $\theta'' = \theta' + \gamma_{\text{offset}}$. This behaviour is better understood in

Fig. 5, where it is possible to see that if there is no offset in the sensor's rotation, $\theta'' = \theta'$.

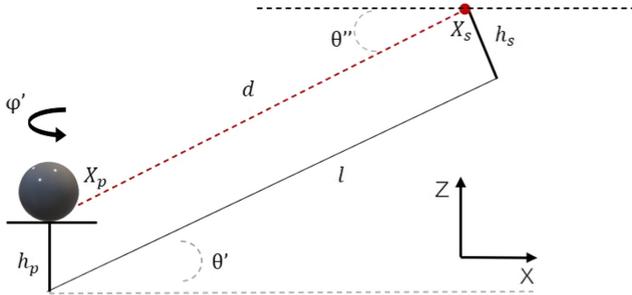


Fig. 5. Updated mechanical design of the proposed system with offsets. XZ plane view.

Finally, to take into account θ'' and the object's offset in the coordinate transformations, it is necessary to make some modifications to the calculations computing X_p .

Therefore, two 2D rotations are made in θ and φ to direct a unit vector, $X_t = (x_t, y_t, z_t)$, according to the sensor beam direction:

$$\begin{cases} x_t = \cos(\varphi')x'_u - \sin(\varphi')z_t \\ y_t = \sin(\varphi')x'_u + \cos(\varphi')z_t \\ z_t = -\sin(\theta'') \end{cases} \quad (4)$$

where $x'_u = -\cos(\theta'')$. After the rotations, the unit vector is scaled by d to obtain the distance of the boundary of the object to the sensor in relation to the center of the plate.

At last, X_p is given by the sum of the sensor's position, X_s , and the scaled rotated unit vector:

$$X_p = X_s + dX_t \quad (5)$$

So, it is intent to optimize the following problem

$$\min \sum_{i=1}^n \|X_{p_i} - X_i\|^2 \quad (6)$$

where X_p , as previously mentioned, represents the scanned points in relation to the center of the support plate, $X = (x, y, z)$ is the nearest point of the original object boundary, and n the number of point on the cloud.

4.2 Optimization

To solve the optimization problem defined on (6), Particle Swarm Optimization (PSO) algorithm was used. This algorithm belongs to Swarm Intelligence

class of algorithms inspired by natural social intelligent behaviors. Developed by Kennedy and Eberhart (1995) [10], this method consists on finding the solution through the exchange of information between individuals (particles) that belong to a swarm. The PSO is based on a set of operations that promote the movements of each particle to promising regions in the search space [3, 4, 10].

This approach was chosen because it has several advantages, according to [12], such as:

- Easy in concept and implementation in comparison to other approaches.
- Less sensitive to the nature of the objective function regarding to other methods.
- Has limited number of hyperparameters in relation to other optimization techniques.
- Less dependent of initial parameters, i.e., convergence from algorithm is robust.
- Fast calculation speed and stable convergence characteristics with high-quality solutions.

However, every approach has its limitations and disadvantages. For instance, authors in [14] stated that one of the disadvantages of PSO is that the algorithm easily fall into local optimum in high-dimensional space, and therefore, have a low convergence rate in the iterative process. Nevertheless, the approach showed good results for our system.

5 Results

To test the approach three case objects were analyzed: Centered sphere, Sphere and Cylinder.

The first and second case were spheres with radius of 0.05 m. The first sphere was centered on the rotating plate where the other had an offset of $X, Y = [0.045, 0.045]$ m. Finally, the third object was a cylinder with $h = 0.08$ m and $r = 0.06$ m.

Since PSO is a stochastic method, 20 independent runs were done to obtain more reliable results. The Table 1 presents the results obtained by PSO, describing the best minimum value obtained on the executed runs, the minimum average from all runs (Av) and their respective standard deviation (SD). To analyze the behavior of the PSO method, the results considering all runs (100%) and 90% of best runs are also presented.

First, the results were very consistent for all the cases as can be verified by the standard deviations, average values, and minima being practically zero. This is also corroborated by comparing the results from all runs (100%) with 90% of the best runs, where the averages from (100%) matched the averages from (90%) of best runs and the standard deviations values maintained the same order of magnitude. Also, another way to look at the consistency is by comparing the best minimum value of executed runs with their respective averages, where all cases match. Still, the only case where the PSO algorithm suffered a little was the

Table 1. PSO results for the three objects

	Minimum	100%		90%	
		Av	SD	Av	SD
Centered Sphere	7.117×10^{-8}	7.117×10^{-8}	5.8×10^{-14}	7.117×10^{-8}	6.06×10^{-14}
Sphere	4.039×10^{-5}	4.039×10^{-5}	1.3×10^{-13}	4.039×10^{-5}	1.18×10^{-13}
Cylinder	3.325×10^{-2}	3.325×10^{-2}	3.2×10^{-7}	3.325×10^{-2}	2.4×10^{-7}

cylinder, where although the minimum tended to 0, it didn't reached the same order of magnitude of the other cases. Therefore, we can confirm that the PSO achieved global optimum in the sphere cases, on the other hand, the cylinder case is dubious. There is also the possibility that the objective function for the cylinder case needs some improvements.

In terms of minimizers, they are described in Table 2. It is possible to see that for the centered sphere, all the parameters are practically 0. This is expected, since the centered sphere did not have any offsets. The same can be said for the cylinder. Finally, for the sphere, it is possible to see that although the sphere had offsets of $X, Y = [0.045, 0.045]$, the minimizers of X, Y are practically zero. This is also expected since as the sphere was displaced on the rotating plate, the sensor acquired data points from the plate itself (on the center). Therefore, the portion of the points representing the sphere was already "displaced" during the acquisition, i.e., the offsets in X, Y were already accounted for during acquisition. By consequence, during transformation to cartesian coordinates. Therefore, the only offset that was impactful on the transformation was the φ_{offset} minimizer, where it accounts for a rotation in the Z axis (to rotate the point cloud to match the object). Of course, the other minimizers had their impact as well during transformation.

Table 2. Minimizers obtained by PSO

	x_{offset} [m]	y_{offset} [m]	φ_{offset} [rad]	θ_{offset} [rad]	γ_{offset} [rad]
Centered Sphere	-4.704×10^{-7}	-5.472×10^{-8}	-5.031×10^{-6}	$+4.953 \times 10^{-5}$	$+2.175 \times 10^{-5}$
Sphere	-7.579×10^{-3}	8.087×10^{-2}	1.566×10^{-1}	-8.885×10^{-5}	-1.661×10^{-4}
Cylinder	5.055×10^{-4}	-3.371×10^{-4}	-2.862×10^{-1}	5.046×10^{-2}	1.057×10^{-1}

The behavior of the PSO approach for the centered sphere, its representation in simulation, and its optimized point cloud over the original object can be seen, respectively, in Figs. 6, 7a, and 7b.

As can be seen from the illustrations, and from Tables 1 and 2, the PSO managed to find the global optimum of the system's parameters. Note that, in Fig. 6, the algorithm converged to the global minimum near iteration 15, nevertheless it ran for more than 70 iterations (note that these iterations are from the PSO algorithm itself, therefore they belong to just one run from the all

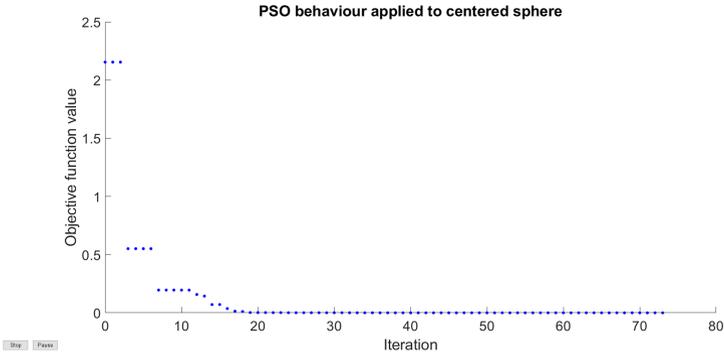
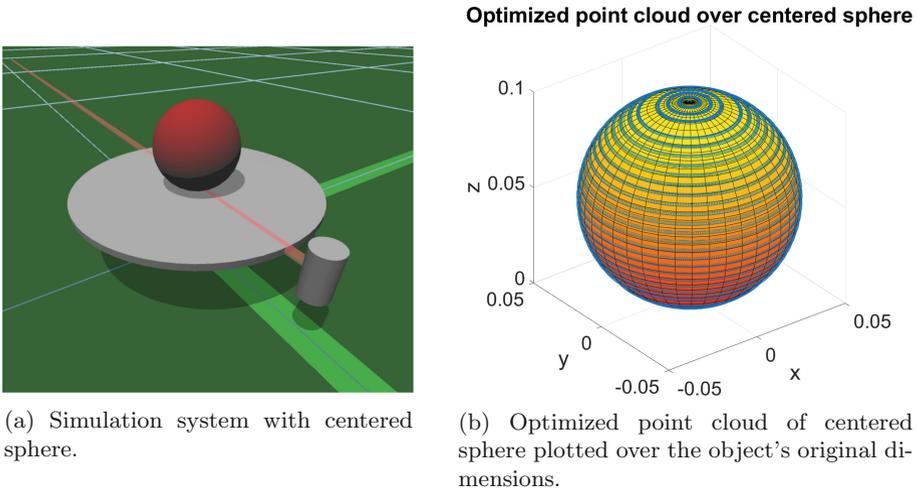


Fig. 6. PSO behavior regarding the centered sphere case.



(a) Simulation system with centered sphere.

(b) Optimized point cloud of centered sphere plotted over the object's original dimensions.

Fig. 7. Simulated system (left). Optimized point cloud (right).

20 executed runs). Thus, the generated point cloud fit perfectly to the original object, as can be seen in Fig. 7b.

In the same reasoning, the behavior of the PSO approach for the cylinder, its system simulation, and its optimized point cloud over the original object can be seen, respectively, in Figs. 8, 9a, and 9b.

It is possible to see in Fig. 8 that after 10 iterations the PSO algorithm converged to the minimum value displayed in Table 1, generating the parameters described in Table 2. The optimized point cloud, however, suffered a tiny distortion on its sides, where it can be seen in Fig. 9b. This probably happened because of rounding numbers. This distortion is mainly caused by the small values of θ_{offset} and γ_{offset} .

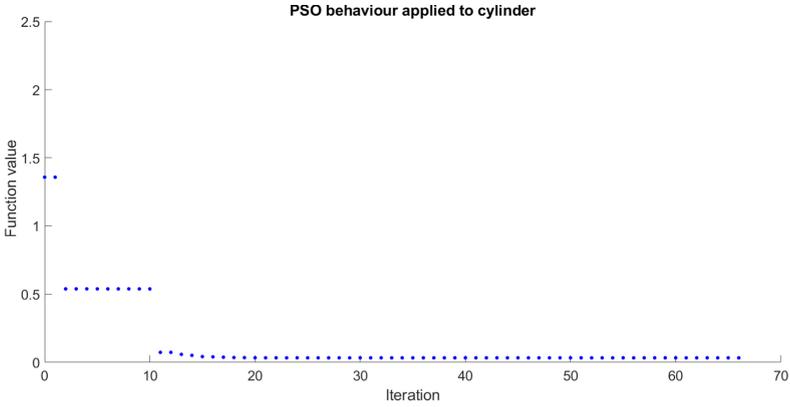


Fig. 8. PSO behavior regarding the cylinder case.

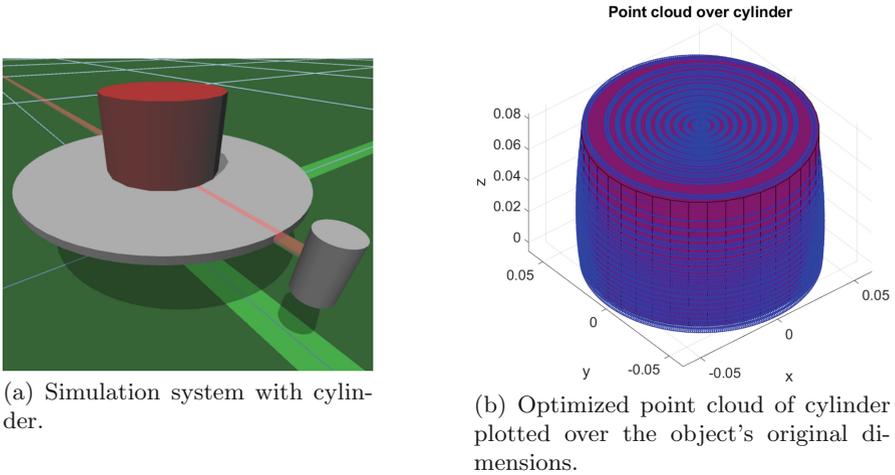


Fig. 9. Simulated system (left). Optimized point cloud (right).

Finally, the behavior of the PSO approach for the sphere with offset, its system simulation, and its optimized point cloud over the original object can be seen, respectively, in Figs. 10, 11a, and 11b.

Same as the centered sphere case, the point cloud fits perfectly over the object with the dimensions from the original object. This is expected, since the PSO managed to find the global optimum of the objective function near iteration 10, generating the optimal parameters for the system, as can be seen in Tables 1 and 2.

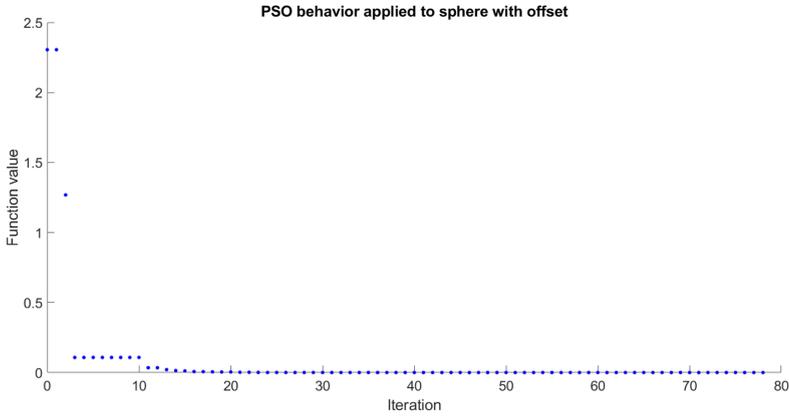


Fig. 10. PSO behavior regarding the sphere with offset case.

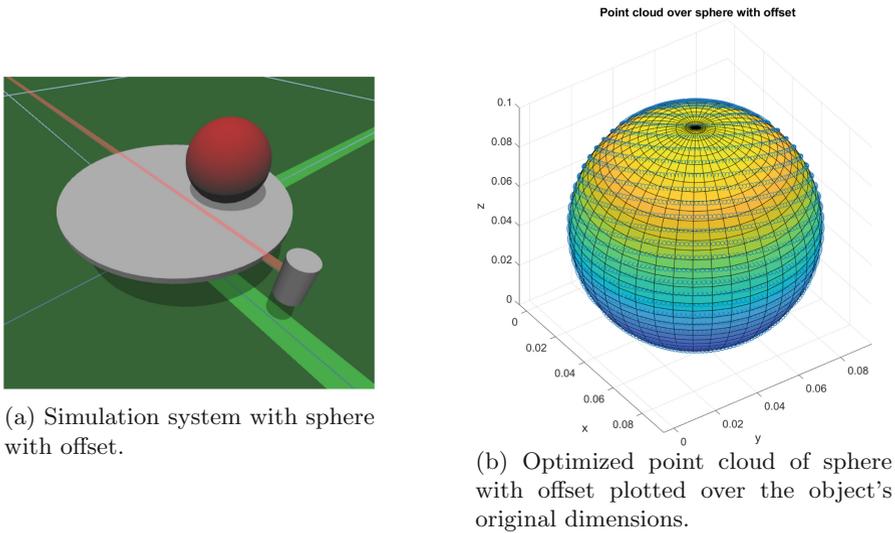


Fig. 11. Simulated system (left). Optimized point cloud (right).

6 Conclusions and Future Work

This work proposed a solution to the imperfections that are introduced when assembling a 3D scanner, due to assumptions that do not typically hold in a real scenario. Some uncertainties associated with angles and offsets, which may arise during the assembly procedures, can be solved resorting to optimization techniques. The Particle Swarm Optimization algorithm was used in this work to estimate the aforementioned uncertainties, whose outputs allowed to minimize the error between the position and orientation of the object and the resulting

final point cloud. The cost function considered the minimization of the quadratic error of the distances which were minimized close to zero in all three cases, validating the proposed methodology. As future work, this optimization procedure will be implemented on the real 3D scanner to reduce the distortions on the scanning process.

Acknowledgements. The project that gave rise to these results received the support of a fellowship from "la Caixa" Foundation (ID 100010434). The fellowship code is LCF/BQ/DI20/11780028. This work has also been supported by FCT - Fundação para a Ciência e Tecnologia within the Project Scope: UIDB/05757/2020.

References

1. Ajbar, W., et al.: The multivariable inverse artificial neural network combined with ga and pso to improve the performance of solar parabolic trough collector. *Appl. Thermal Eng.* **189**, 116651 (2021)
2. Arbutina, M., Dragan, D., Mihic, S., Anisic, Z.: Review of 3D body scanning systems. *Acta Tech. Corviniensis Bulletin Eng.* **10**(1), 17 (2017)
3. Bento, D., Pinho, D., Pereira, A.I., Lima, R.: Genetic algorithm and particle swarm optimization combined with powell method. *Numer. Anal. Appl. Math.* **1558**, 578–581 (2013)
4. Bratton, D., Kennedy, J.: Defining a standard for particle swarm optimization. *IEEE Swarm Intell. Sympo.* (2007)
5. Braun, J., Lima, J., Pereira, A., Costa, P.: Low-cost 3d lidar-based scanning system for small objects. In: 22^o International Conference on Industrial Technology 2021. *IEEE proceedings* (2021)
6. Franca, J.G.D.M., Gazziro, M.A., Ide, A.N., Saito, J.H.: A 3d scanning system based on laser triangulation and variable field of view. In: *IEEE International Conference on Image Processing 2005*. vol. 1, pp. I-425 (2005). <https://doi.org/10.1109/ICIP.2005.1529778>
7. Ghorbani, E., Moosavi, M., Hossaini, M.F., Assary, M., Golabchi, Y.: Determination of initial stress state and rock mass deformation modulus at lavarak hepp by back analysis using ant colony optimization and multivariable regression analysis. *Bulletin Eng. Geol. Environ.* **80**(1), 429–442 (2021)
8. He, Z., Shi, T., Xuan, J., Jiang, S., Wang, Y.: A study on multivariable optimization in precision manufacturing using mopsonns. *Int. J. Precis. Eng. Manuf.* **21**(11), 2011–2026 (2020)
9. Jin, C., Li, S., Yang, X.: Adaptive three-dimensional aggregate shape fitting and mesh optimization for finite-element modeling. *J. Comput. Civil Eng.* **34**(4), 04020020 (2020)
10. Kennedy, J., Eberhart, R.: Particle swarm optimization. *IEEE International Conference on Neural Network*, pp. 1942–1948 (1995)
11. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2252–2261 (2019)
12. Lee, K.Y., Park, J.B.: Application of particle swarm optimization to economic dispatch problem: advantages and disadvantages. In: *2006 IEEE PES Power Systems Conference and Exposition*, pp. 188–192 (2006). <https://doi.org/10.1109/PSCE.2006.296295>

13. Lempitsky, V., Boykov, Y.: Global optimization for shape fitting. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007). <https://doi.org/10.1109/CVPR.2007.383293>
14. Li, M., Du, W., Nian, F.: An adaptive particle swarm optimization algorithm based on directed weighted complex network. *Math. Probl. Eng.* 2014 (2014)
15. Ma, T.: Filtering adaptive tracking controller for multivariable nonlinear systems subject to constraints using online optimization method. *Automatica* **113**, 108689 (2020)
16. Rehman, W.U., et al.: Model-based design approach to improve performance characteristics of hydrostatic bearing using multivariable optimization. *Mathematics* **9**(4), 388 (2021)
17. Soltani, S., et al.: The implementation of artificial neural networks for the multivariable optimization of mesoporous nio nanocrystalline: biodiesel application. *RSC Advances* **10**(22), 13302–13315 (2020)
18. Straub, J., Kading, B., Mohammad, A., Kerlin, S.: Characterization of a large, low-cost 3d scanner. *Technologies* **3**(1), 19–36 (2015)
19. Swathi, A.V.S., Chakravarthy, V.V.S.S.S., Krishna, M.V.: Circular antenna array optimization using modified social group optimization algorithm. *Soft Comput.* **25**(15), 10467–10475 (2021). <https://doi.org/10.1007/s00500-021-05778-2>
20. Wang, W., Li, Y., Hu, B.: Real-time efficiency optimization of a cascade heat pump system via multivariable extremum seeking. *Appl. Thermal Eng.* **176**, 115399 (2020)