3D printing part orientation optimization: discrete approximation of support volume

Juan C. Guacheta Alba, Sebastian Gonzalez Garzon, Diego A. Nunez, Mauricio Mauledoux, Oscar F. Aviles
DAVINCI Research Group, Mechatronics Engineering Department, Universidad Militar Nueva Granada, Colombia

Article Info

Article history:

Received Sep 13, 2021
Revised Jun 25, 2022
Accepted Jul 18, 2022

Keywords:

3D printing
Evolutionary operation
Multivariable optimization
Part orientation
Random walk
Support volume

ABSTRACT

In three-dimensional (3D) printing, due to the geometry of most parts, it is necessary to use extra material to support the manufacturing process. This material must be discarded after printing, so its reduction is essential to minimize manufacturing time and cost. An important parameter that must be defined before starting the printing process is the part orientation, which has repercussions on the quality, deposition path, and post-processing among others. Usually, the user sets up this parameter arbitrarily, so this paper takes advantage of it on optimization techniques and proposes an approximation of the volume be covered by the support material, which depends directly on the angle of the part to be printed and its geometry. Among mono-objectives optimization strategies, this work focuses on five of them. Their performance is compared by two metrics: support volume and execution time. Then, the best result is compared with commercial software.

Corresponding Author:

Juan C. Guacheta Alba
DAVINCI Research Group, Mechatronics Engineering Department, Universidad Militar Nueva Granada
Carrera 11 # 101-80, Bogotá D. C., Colombia
Email: est.juan.guacheta@unimilitar.edu.co

1. INTRODUCTION

Additive manufacturing (AM), or more commonly known as 3D printing, consist of manufacturing parts employing layer-by-layer material deposition, covering the solid volume. The movements made by the printing nozzle are linked to the piece structure and orientation, and these determine the mechanical behavior of the part and its texture and quality [1], [2]. The part's orientation is an initial parameter for its manufacturing and is commonly set by the user. However, this variable is associated with characteristics of the final product such as the model precision, the number of supports required, and the processing time for its production [3], [4], as well as crucial end-user criteria, such as aesthetics, smoothness, material cost, and energy spent in manufacturing [5]–[8]. Some issues related to wrong printing orientation are volumetric errors that deform the part, high presence of the staircase effect represented in poor surface quality, high construction time [4], material consumption [8], [9], anisotropy [10], and cylindricity and flatness errors [11]–[13].

3D printing has capability to manufacture any geometry compared to other manufacturing processes. However, it is necessary for several parts to print support structures that guarantee structural stability and avoid the collapse or deformation of the material in the regions with overhangs in the manufacturing process [9], [14]. This support is eventually represented in waste material, additional costs [15], and possible defects on the surfaces [9]. There are support materials that can be removed chemically, improving the result of the part [14]. However, it is an additional process that affects the manufacturing time and cost. This material is directly related to the support volume, which corresponds to the region used to construct the holdup structure.

Some aspects such as: concavity, geometric shape, size, and islands must be considered to define and obtain the support volume of the parts to be manufactured [16]. To treat and analyze the support volume, it is crucial to consider a continuous but non-smooth function concerning the orientation angles [17]. Its behavior will be defined by the geometry of the part to be manufactured. Several authors propose strategies to obtain or approximate the support volume, such as the use of the kth nearest point algorithm [16], convex hull surface triangles method [13], or a Quadtree decomposition [8] to find the volume of support structures. Another essential aspect considered in the aforementioned strategies is the minimum self-support angle, which is suggested for direct metal laser sintering (DMLS) printed parts of 45 degrees [14].

The above considerations have been a topic of study to optimize objective functions to improve the printing process, being the principal applied function of the support volume. Some mono-objective optimization techniques that have been applied to this problem are particle swarm optimization (PSO) [7], ray-tracing method [13], optimization methods like electromagnetism [4], or perceptual models [9]. These techniques become complex in their programming and implementation but guarantee to obtain global solutions to the problem. Along with the support volume, multi-objective techniques have been applied to optimize more objective functions, such as resolution error [5], [11], surface roughness [18], printing time, and the number of aggregate suspensions [3]. The main multi-objective techniques used to solve this optimization problem are genetic algorithms [3] and particle swarm optimizers [15].

Due to this, there is no evidence that focuses on optimizing the orientation of pieces that guarantees a lower volume of supports. This work presents a solution using multivariate optimization techniques. This paper is organized as follows: section 2 defines the method used to minimize the support material, employing a correct orientation in printing pieces and multivariate optimization algorithms. Section 3 presents the results obtained on a set of pieces. Also, the performance of the applied algorithms is compared. Finally, section 4 points out the recommended algorithms in multivariate problems and exposes a discrete approximation validation compared with commercial software. The following sections expose the capacity of the discrete approximation of continuous problems and the feasibility of applying simple multivariate optimization algorithms in problems with application in additive manufacturing, saving costs and time in manufacturing pieces.

2. METHOD

The main objective of this work is to present an algorithm for optimizing the piece orientation. This process requires several steps described below: first, it is required to define the decision variables that correspond to the rotation angles of any piece in STL format. Second, the mathematical formulation and programming of the objective function to be optimized, corresponding to the approximation of the volume used to print the support material required in the printing. Third, the five multivariate optimization algorithms used to solve the formulated problem are exposed as the performance metrics. At the end of this chapter, the case study section presents pieces used to assess the algorithms and metrics.

2.1. Problem definition

To reduce the deposited material in the 3D printing process, the minimization of the support volume of the parts, which is directly related to the deposited support material, is proposed. To structure the problem, the three angles of rotation roll (ψ), pitch (θ) and yaw (φ), also called navigation angles or Tait-Bryan angles, are defined as decision variables. These are necessary to obtain all possible orientations of the piece. The rotation matrix used on the part, which represents a rotation in space, is the matrix R_{RPY} defined in (1), where R_z, R_y and R_x are the matrix rotation at the z, y and x axis, respectively.

$$R_{RPY} = R_{z,\varphi}\, R_{y,\theta}\, R_{x,\psi} \tag{1}$$

With the three decision variables defined, the discrete approximation of the support volume is defined, which corresponds to the objective function to be minimized. The discrete approximation of the support volume is performed with the three defined decision variables, which corresponds to the objective function to be minimized. When the part is obtained and the rotation matrix is applied, an octahedral mesh of thickness d is created, completely covering the part. The mesh size affects the computing time and the precision of the approximation of the support volume. The mesh is swept in the z-direction, and at each point it is evaluated whether it corresponds to a region to be supported, considering the 45-degree rule for overhangs [14].

The pseudocode of this objective function is shown in Algorithm 1. Although it is possible with two angles to find the orientations for printing, the discrete approach used to obtain the support volume is decided to use the three angles. For selecting search methods and algorithms applied to the proposed problem, its formal definition is made, which is an unconstrained single-objective multivariable optimization problem. This problem is focused on its application on 3D printers and is mathematically represented in (2), where x

corresponds to the angles roll (ψ), pitch (θ) and yaw (φ), while $f(x)$ corresponds to the approximate support volume. Although the desired solution is sought in a range from $-\pi$ to π, they are not constraints to allow transitions between quadrants.

$$min\, f(x) \;\therefore\; x = \{\varphi, \theta, \psi\} \in S \subseteq \mathbb{R}^3 \tag{2}$$

Algorithm 1. Objective function pseudocode

```
Procedure Support volume approximation
Input:  Piece; φ; θ; ψ; d.
Output: Volume.
Rotation of the piece using matrix R_RPY
Create the mesh with thickness d
covering the piece
Location of mesh points inside the piece
Z-direction mesh sweep
for each Point in meshing
     if Point is not inside the piece and Point
     has no support at 45 degrees then
        Point required for support
TotalPoints = sum(Necessary support points)
Volume ← TotalPoints * d ³
```

In Figure 1, the massive overhang test by Thingster is used to evaluate the performance of the designed discrete objective function and its frame of reference is presented, as well as the rotation angles. It can be observed that the volume covered with support material is not calculated for print angles less than 45 degrees. In contrast, for angles greater than this angle, this volume is approximated by the lower mesh. The region of the mesh that approximates the support volume on the parts used in this paper is shown in green.

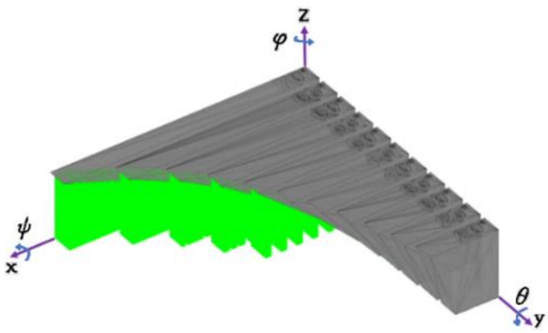


Figure 1. Discrete volume approximation of support material in massive overhang test

2.2. Optimization algorithms

In this section, five algorithms are executed. It is to analyze and compare solutions found and evaluate the performance of multivariate algorithms. The algorithms used are described below.

2.2.1. Random walk

The random walk is a search method that describes a path that includes a succession of random steps in the mathematical space [19]. This method generates a sequence of approximations by a unit random vector generated at the *i-th* step. A random angle λu is generated if the *i-th* member of the group is chosen as the wanderer of the *z-th* iteration [20]. Algorithm 2 exposes the pseudocode used for random walk scheduling, defining parameters a step length, λ is set to π, a minimum allowable step length, ε is set to 1×10^{-4}, and a maximum allowable number of iterations, N is set to 20.

Algorithm 2. Random walk pseudocode

```
Procedure Random walk
Input:  f(x); x0; λ; ε; N.
Output: xmin
f0 ← f(x0)
i ← 0
u ← random unit vector of n decision variables
x1 ← x0 + λ u
f1 ← f(x)
if f1<f0 then
       x0 ← x1
        f0 ← f1
       goto 2
else
  if i<N then
        i←i+1
        goto 3
  Else
        λ←λ/2
        if λ≤ε then
               xmin ← x0
               exit;
               Else
                      goto 3
```

2.2.2. Evolutionary operation

The evolutionary operation (EVOP), introduced by George Box, is considered in applications where only one output feature is optimized [21]. This method requires (2N+1) points, where N is the number of decision variables, of which $2N$ are the corners of a hypercube. The (2N+1) function values are compared, and the best point is identified [22]. If, in any iteration, the current point is not improved, then the size of the hypercube is reduced. Algorithm 3 shows the pseudocode used for EVOP programming, defining as parameters a tolerance $\varepsilon \rightarrow 1\times10^{-4}$, and a size reduction vector $\Delta \rightarrow [1,1,1]$ for the decision variables $[\varphi, \theta, \psi]$.

Algorithm 3. Evolutionary operation pseudocode

```
Procedure Evolutionary operation
Input: f(x); x_0; Δ; ε; N.
Output: x_min
x̄ ← x_0
if ||Δ|| < ε then
        x_min ← x̄
        exit
else
        create 2^N points by adding and
subtracting Δ_i/2 from each variable at
point x̄
calculate f(x) in the (2^N+1) points
x̄ ← point with the lowest function value

if x̄ = x_0 then
        Δ_i ← Δ_i / 2
        goto 2
else
        x_0 ← x̄
        goto 2
```

2.2.3. Simplex method

It is the search method proposed by Spendley, Hext, and Himsworth and later improved by Nelder and Mead, which seeks to minimize continuous and multidimensional unconstrained optimization problems [23]. A regular simplex is a polyhedron composed of $(N+1)$ equidistant points, which form its vertices, where N is the number of decision variables. The main objective of the method is the generation of a new simplex by projecting any vertex at an appropriate distance through some movements such as reflection (α), expansion (γ), and contraction (β) [24]. Algorithm 4 details the operations and conditions necessary to apply each of these movements. For the case study, the following parameters were used: $\alpha \rightarrow 0.6$, $\gamma \rightarrow 1.3$, $\beta \rightarrow 0.7$ and a tolerance $\varepsilon \rightarrow 100$, related to the objective function in the scale of million.

Algorithm 4. Simplex method pseudocode

```
Procedure Simplex
Input: f(x); x_0; α; γ >1; β ∈ (0,1); ε; N.
Output: x_min
Find x_h (worst point), x_l (best point), and x_g
(second worst point)
x_c ← (1/N) Σ_{i=1,i≠h}^{N+1} x_i (centroid)
x_r ← 2x_c−x_h (reflection)
x_new ← x_r
if f(x_r)<f(x_l) then
        x_new ← (1+γ) x_c − x_h (expansion)
else if f(x_r)≥f(x_h) then
        x_new ← (1 −β) x_c+β x_h (contraction inside)
else if f(x_g)<f(x_r)<f(x_h) then
        x_new ← (1+β)x_c−βx_h (contraction
outside)
Calculate f(x_new)
x_h ← x_new
Q ← [Σ_{i=1}^{N+1} (f(x_i)−f(x_c))^2 / (N+1)]^(1/2)
if Q<ε then
        x_min ← x_l
        exit
else
        goto 1
```

2.2.4. MATLAB fmincon function

As a comparison method for programming, the MATLAB fmincon function performs the search for the minimum of a nonlinear multivariable scalar function, with or without restrictions. Like the three previously mentioned, this method needs an initial point to be executed, so the same point is used for all algorithm executions. The fmincon function allows the selection of five different algorithms, so the default method interior-point was selected [25].

2.2.5. MATLAB ga function

Considering the algorithms investigated, the MATLAB ga function is used, which corresponds to a genetic algorithm used to find the global minimum in highly nonlinear problems. In turn, it is based on a natural selection process that mimics biological evolution and is applied on constrained and unconstrained optimizations. Unlike classical algorithms, it generates a population of points for each iteration, and its calculation uses random number generators [25].

2.3. Performance metrics

On each execution carried out, the following evaluation metrics are measured, which are grouped by the algorithm for comparison:

- Approximate part support volume: this metric corresponds to the objective function described in Algorithm 1 and is expressed in mm^3. A weighted average is performed on each implemented algorithm for its evaluation, using (3), where x represents the approximate volume obtained for each part. In contrast, w represents the inverse of the total volume of the part mesh, which is represented as a percentage of deposited material for the mesh used.
- Algorithm execution time: this time is captured using a stopwatch timer and was taken only from the algorithm, without considering the part reading or variable initialization, and is expressed in seconds. A weighted average ($\bar{x}$) is performed using (3), where x corresponds to the estimated time and weights w corresponds to the inverse of the part mesh size.

$$\bar{x} = \frac{\sum x_i w_i}{\sum w_i} \tag{3}$$

2.4. Case study

A set of parts is selected to apply the objective function and algorithms mentioned. Figure 2 shows the set of the parts used to evaluate the performance of the orientation optimization process, whose names listed from left to right are: Wingnut_6x9 by Mike_mattala, Frog ring by Hlebushek2187, 3DBenchy by Creative Tools, measuring cup 20 ml by Ndl, Samsung Galaxy Watch Stand by Lars_kglr, Simple bunny ball joint doll by Dollightful, Smol Kitchen/Hobby Funnel by Towerdweller and Geo Cube by Burtronix. This group collects pieces with simple and complex geometries, reflected in flat and curved surfaces and concave and convex surfaces, to test the support volume estimation. All these pieces are licensed under the creative commons licenses for use and sharing.

The mesh used to approximate the geometry has an average of 1.3 million points, being these cubes with an average volume of 0.7 mm^3. The programming and execution are performed in MATLAB R2021a software. These tests were performed on a computer with an Intel core™ i5-10300H processor and installed RAM of 8 GB. The multivariate optimization methods applied to the exposed geometries are the five algorithms mentioned in the previous section, so eight executions were performed per algorithm.

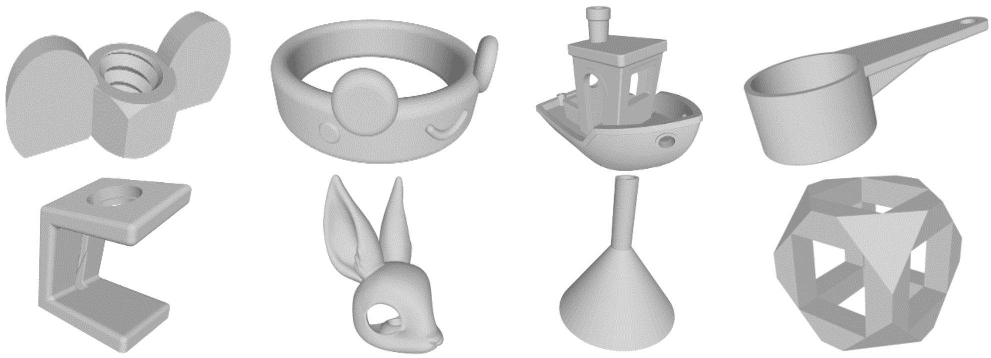


Figure 2. Piece designs used for optimization

3. RESULTS AND DISCUSSION

Once the problem, the algorithms, and the case study to be used are formulated, the executions of the five algorithms for the eight pieces are carried out. For each of them, the optimal orientation solution in roll (ψ), pitch (θ) and yaw (φ) is obtained, as well as the measurement of the approximate support volume of the piece and the execution time. The orientations obtained in all the executions are grouped to analyze this data thoroughly. Table 1 shows the best solutions found by the optimization algorithms used. These solutions are expressed in the range from -π to π, covering all possible orientations presented by the parts. It is worth mentioning that, for the ‘Smol kitchen’ piece, the simplex method was the one that found the optimal solution, while for the ‘Measuring cup’ and ‘Frog ring’ pieces, the optimal solution was found by the random walk method. The evolutionary operation algorithm obtained the remaining optimal solutions, which found the highest number of best solutions. The generic MATLAB algorithms were able to solve the optimization problem correctly, but not with the precision of the first three.

Continuing with analyzing the optimal orientations obtained for the eight pieces, a visual evaluation of the solutions is carried out. Figure 3 shows a set of with the optimal support volume approximations for four selected parts. These solutions show that it solves orientations that visually can be selected for printing, such as the ‘Frog ring’ piece exposed in Figure 3(a). In addition, it also correctly solves other parts, such as the ‘Bunny's head’, shown in Figure 3(b), which has a hollow presence in its geometry, and large curved sections, offering the user a solution that is not easy to obtain. In addition, in Figures 3(c) and 3(d), we can see the prevalence of the 45-degree rule for printing, where a defined orientation for a flat face in these cases results in greater volume and support material.

Table 1. Optimal orientation results for pieces

Pieces	φ [rad]	θ [rad]	ψ [rad]	Volume [mm^3]
Wingnut	2.5743	0	-0.0012	3.5641
Frog ring	0.2076	3.1414	3.1415	7.8105
Smol kitchen	0.4524	0.0012	0.0273	1688.9
3Dbenchy	1.3750	0.5406	-0.4528	857.31
Samsung stand	0.5623	0.8740	0.0464	2468.4
Bunny's head	-1.6258	-0.0005	-2.3223	879.028
Measuring cup	-1.4447	2.3620	1.2969	370.012
Geo cube	0.2559	0	0.3341	1698.4

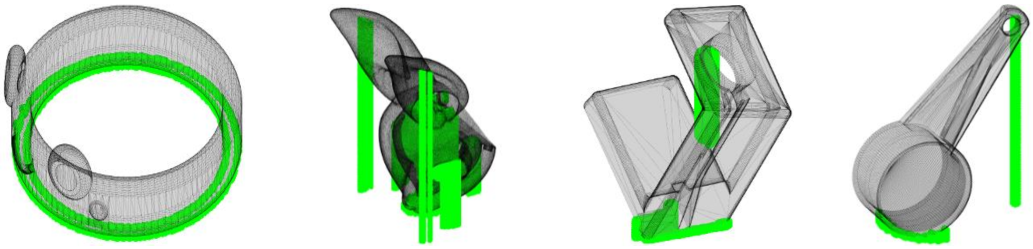

(a) (b) (c) (d)

Figure 3. Support volume in optimal piece orientations (a) frog ring, (b) Samsung stand, (c) measuring cup, and (d) bunny's head.

To analyze the accuracy and speed of the algorithms, Figure 4 shows their performance curves in the search for the orientation of the ‘Geo cube’ piece. The evolutionary operation and random walk algorithms, on this optimization problem, have a faster convergence, also with a high accuracy on the first iterations. Meanwhile, it is shown that the simplex algorithm finds with good accuracy the optimal solution, but in a long time. On the other hand, the MATLAB fmincon function has low precision and the MATLAB ga function consumes a high computation time per iteration.

Once the results related to the orientations obtained by the algorithms that most effectively minimized the support volume have been analyzed, each algorithm's efficiency on all the pieces is analyzed. For this, the evaluation metrics defined in section 2.4 are used: the approximate support volume of the part and the algorithm time. It denotes that the weighted average is made using (3) on each algorithm for the set of pieces. Table 2 shows the results of the metrics mentioned. These result from the weighted average for all the pieces on each algorithm. The evolutionary operation search method found better solutions to the selected parts for the approximate support volume metric, outperforming the simplex method and the MATLAB genetic algorithm. The better result of the direct search methods is presented because the optimization problem does not become complex and allows applying techniques based on the direct search. In addition, being a discrete approximate function, the evolutionary operation algorithm facilitates its implementation and execution.

On the other hand, the runtime metric for each algorithm performs better with MATLAB fmincon function. However, these results are variant by part, as some are solved efficiently, while others do not converge quickly. All algorithms consume a similar execution time except for the MATLAB genetic algorithm, which takes 20 times longer due to its nature and the number of evaluations required on the objective function. Finally, to corroborate the correct approximation of the volume to be coated required by the support material, the open software PrusaSlicer is used, which exposes the printed material for the manufacture of parts graphically. Figure 5 shows the approximation performed on the ‘3Dbenchy’ part in Figure 5(a) and after, a support material by PrusaSlicer in Figure 5(b), presenting a similar result in both cases. Because it is an approximation, the programmed objective function does not consider the necessary support on the front holes of the part. However, it works correctly for orientation optimization, and the discrete objective function is valid.

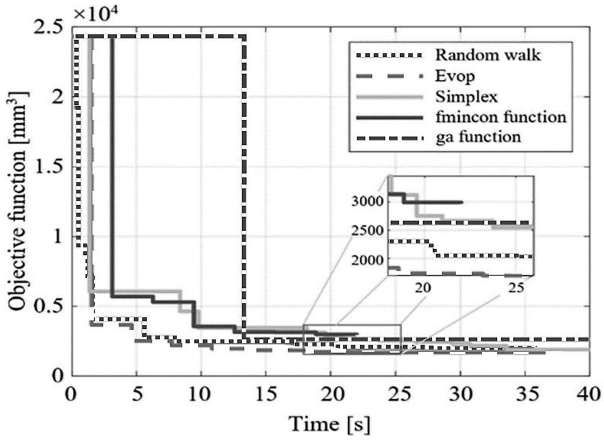
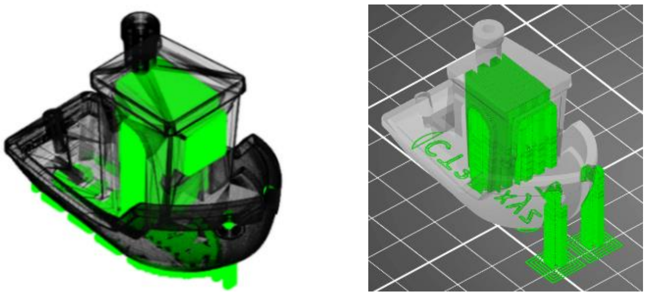


Figure 4. Performance of the algorithms used on Geo cube

Table 2. Weighted results of volume and computation time metrics for all parts

Optimization techniques	Volume [mm^3]	Time [s]
Random walk	135.5765	46.9499
Evolutionary operation	100.6915	45.1587
Simplex method	126.3512	44.5598
fmincon by MATLAB	194.9746	43.8417
GA by MATLAB	132.5944	947.53



(a) (b)

Figure 5. Support material used for '3Dbenchy' printing (a) objective function and (b) PrusaSlicer

4. CONCLUSION

Direct search methods are not commonly used due to their rapid convergence into local solutions. However, they can be applied to unrestricted problems with a finite search range and achieve good results. Comparing the performance of the executed algorithms, it was observed that the random walk and operational evolution methods were able to find the global minimum on the orientation optimization problem. In contrast to this, the simplex method found local solutions, verified by the genetic algorithm used. On the support volume optimization problem, the operational evolution method was the one that obtained the best results, and with an execution time like that obtained by the MATLAB fmincon function. Simple search methods were applied, which have a low computational cost compared to bio-inspired algorithms, which on the case study analyzed, take up to 20 times longer. The operational evolution method converges on the same global minimum as the genetic algorithm with higher accuracy.

The approximation of the support volume in a discrete way allows to calculate this variable to any piece, regardless of its geometry, making this process automatic and simple, to comparison of its mathematical formulation. The effectiveness of this approach is corroborated by a commercial 3D printing software, where the regions needed for support printing correspond to the discrete support volume. Unlike other research, non-symmetric geometries such as 'Samsung stand' are used, so the search surface is more extensive and presents a unique global minimum. Finally, it is recommended to use optimization strategies in the preprocessing phase in 3D printing, to select the best orientation to parts in which it is not possible visually, allowing to reduce costs and time.

Since the present proposed algorithm automatically selects the optimal print orientation, it seeks to make a preprocessing software that offers the best part orientation to the user before printing his part. This process ensures that it consumes less printing material, which is reflected in the cost and contamination in the manufacturing process. In addition, the use of GPUs to parallelize the discrete objective function is proposed since its computation time can be reduced with this strategy. Finally, it would be interesting to apply multi-objective optimization techniques having more objective functions, such as the staircase effect, printing time or surface finish, and other decision variables such as the printing pattern.

ACKNOWLEDGEMENTS

This paper is a research product from the IMP-ING-3122 high-impact project funded by the Research Vice-Chancellor of Universidad Militar Nueva Granada–2021. *Producto derivado del proyecto IMP-ING-3122 financiado por la Vicerrectoría de Investigaciones de la Universidad Militar Nueva Granada –Vigencia 2021*.

REFERENCES

[1] L. Di Angelo, P. Di Stefano, and A. Marzola, "Surface quality prediction in FDM additive manufacturing," *The International Journal of Advanced Manufacturing Technology*, vol. 93, no. 9–12, pp. 3655–3662, Dec. 2017, doi: 10.1007/s00170-017-0763-6.

[2] J. Mueller and K. Shea, "The effect of build orientation on the mechanical properties in inkjet 3D-printing," in *Twenty-Sixth Annual International Solid Freeform Fabrication (SFF) Symposium – An Additive Manufacturing Conference, Austin, United States*, 2015, pp. 983–992.

[3] Z. Li, A. Hu, J. Fu, X. Wu, and H. Li, "Printing orientation optimization of 3D model," in *Proceedings of the 2nd International Conference on Computer Science and Application Engineering - CSAE '18*, 2018, pp. 1–5, doi: 10.1145/3207677.3278034.

[4] M. A. Matos, A. M. A. C. Rocha, and A. I. Pereira, "Improving additive manufacturing performance by build orientation optimization," *The International Journal of Advanced Manufacturing Technology*, vol. 107, no. 5–6, pp. 1993–2005, Mar. 2020, doi: 10.1007/s00170-020-04942-6.

[5] J. Son and S. Choi, "Orientation selection for printing 3D models," in *2015 International Conference on 3D Imaging (IC3D)*, Dec. 2015, pp. 1–6, doi: 10.1109/IC3D.2015.7391823.

[6] W. M. Wang, C. Zanni, and L. Kobbelt, "Improved surface quality in 3D printing by optimizing the printing direction," *Computer Graphics Forum*, vol. 35, no. 2, pp. 59–70, May 2016, doi: 10.1111/cgf.12811.

[7] M. Barclift, A. Armstrong, T. W. Simpson, and S. B. Joshi, "CAD-integrated cost estimation and build orientation optimization to support design for metal additive manufacturing," Aug. 2017, doi: 10.1115/DETC2017-68376.

[8] P. Das, R. Chandran, R. Samant, and S. Anand, "Optimum part build orientation in additive manufacturing for minimizing part errors and support structures," *Procedia Manufacturing*, vol. 1, pp. 343–354, 2015, doi: 10.1016/j.promfg.2015.09.041.

[9] X. Zhang, X. Le, A. Panotopoulou, E. Whiting, and C. C. L. Wang, "Perceptual models of preference in 3D printing direction," *ACM Transactions on Graphics*, vol. 34, no. 6, pp. 1–12, Nov. 2015, doi: 10.1145/2816795.2818121.

[10] J. F. P. Lovo, C. A. Fortulan, and M. M. da Silva, "Optimal deposition orientation in fused deposition modeling for maximizing the strength of three-dimensional printed truss-like structures," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, vol. 233, no. 4, pp. 1206–1215, Mar. 2019, doi: 10.1177/0954405418774603.

[11] R. Paul and S. Anand, "Optimization of layered manufacturing process for reducing form errors with minimal support structures," *Journal of Manufacturing Systems*, vol. 36, pp. 231–243, Jul. 2015, doi: 10.1016/j.jmsy.2014.06.014.

[12] G. Moroni, W. P. Syam, and S. Petrò, "Functionality-based part orientation for additive manufacturing," *Procedia CIRP*, vol. 36, pp. 217–222, 2015, doi: 10.1016/j.procir.2015.01.015.

[13] M. P. Zwier and W. W. Wits, "Design for additive manufacturing: Automated build orientation selection and optimization," *Procedia CIRP*, vol. 55, pp. 128–133, 2016, doi: 10.1016/j.procir.2016.08.040.

[14] A. T. Gaynor and J. K. Guest, "Topology optimization considering overhang constraints: Eliminating sacrificial support material in additive manufacturing through design," *Structural and Multidisciplinary Optimization*, vol. 54, no. 5, pp. 1157–1172, Nov. 2016, doi: 10.1007/s00158-016-1551-x.

[15] H. Shen, X. Ye, G. Xu, L. Zhang, J. Qian, and J. Fu, "3D printing build orientation optimization for flexible support platform," *Rapid Prototyping Journal*, vol. 26, no. 1, pp. 59–72, Jan. 2020, doi: 10.1108/RPJ-09-2018-0252.

[16] M. A. Habib and B. Khoda, "Attribute driven process architecture for additive manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 44, pp. 253–265, Apr. 2017, doi: 10.1016/j.rcim.2016.10.003.

[17] B. Ezair, F. Massarwi, and G. Elber, "Orientation analysis of 3D objects toward minimal support volume in 3D-printing," *Computers & Graphics*, vol. 51, pp. 117–124, Oct. 2015, doi: 10.1016/j.cag.2015.05.009.

[18] S. E. Brika, Y. F. Zhao, M. Brochu, and J. Mezzetta, "Multi-objective build orientation optimization for powder bed fusion by laser," *Journal of Manufacturing Science and Engineering*, vol. 139, no. 11, Nov. 2017, doi: 10.1115/1.4037570.

[19] F. Xia, J. Liu, H. Nie, Y. Fu, L. Wan, and X. Kong, "Random walks: A review of algorithms and applications," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 2, pp. 95–107, Apr. 2020, doi: 10.1109/TETCI.2019.2952908.

[20] Z. Li and Z. Li, "A slicing parameter optimization method using group search optimization algorithm in STL model for 3D printing application," *Academic Journal of Engineering and Technology Science*, vol. 2, no. 3, pp. 73–84, 2019, doi: 10.25236/AJETS.020057.

[21] K. Rutten, J. De Baerdemaeker, J. Stoev, M. Witters, and B. De Ketelaere, "Constrained online optimization using evolutionary operation: A case study about energy-optimal robot control," *Quality and Reliability Engineering International*, vol. 31, no. 6, pp. 1079–1088, Oct. 2015, doi: 10.1002/qre.1662.

[22] C. A. C. Coello, *Introduction to evolutionary computing*, (in Spanish), 2nd ed. Computación Evolutiva, Mexico: Academia Mexicana de Computación, 2019.

[23] S. Zapotecas-Martínez and C. A. C. Coello, "MONSS: A multi-objective nonlinear simplex search approach," *Engineering Optimization*, vol. 48, no. 1, pp. 16–38, Jan. 2016, doi: 10.1080/0305215X.2014.992889.

[24] S. Z. Martinez, A. A. Montano, and C. A. C. Coello, "A nonlinear simplex search approach for multi-objective optimization," in *2011 IEEE Congress of Evolutionary Computation (CEC)*, Jun. 2011, pp. 2367–2374, doi: 10.1109/CEC.2011.5949910.
[25] I. The MathWorks, "Optimization toolbox™ user's guide," 2021. https://www.mathworks.com (accessed Jul. 23, 2021).

BIOGRAPHIES OF AUTHORS

Juan C. Guacheta Alba received his B.S. in mechatronics engineering from Universidad Militar Nueva Granada in 2020. Currently, he is working toward the M.Sc. degree in Mechatronics Engineering at Universidad Militar Nueva Granada. His research interests include robotics, multi-agent systems and optimization. He can be contacted at email: est.juan.guacheta@unimilitar.edu.co.

Sebastian Gonzalez Garzon received his B.S. in mechatronics engineering from Universidad Militar Nueva Granada in 2020. Currently, he is working toward the M.Sc. degree in Mechatronics Engineering at Universidad Militar Nueva Granada, Colombia. His research interests include optimization and robotics. He can be contacted at email: est.sebastian.gonz1@unimilitar.edu.co.

Diego A. Nunez received his B.S. in mechatronics engineering from Universidad Militar Nueva Granada in 2005 and his M.Sc. in mechanical engineering from Universidad de los Andes in 2014. He is specialist in polymer processing and is currently working toward a Ph.D. in applied science at Universidad Militar Nueva Granada. His research interests include parallel robots, optimization, and additive manufacturing. He can be contacted at email: danvmoldes@gmail.com.

Mauricio Mauledoux received his B.S. in mechatronics engineering from Universidad Militar Nueva Granada, in 2005. In 2011, he received his PhD degree in Mathematical models, numerical methods, and software systems (Red Diploma) from St. Petersburg State Polytechnic University, Russia. In 2012, he joined the Department of Mechatronic Engineering, at Universidad Militar Nueva Granada, in Colombia as Assistant Professor. His current research interests include robotics, automatic control, multi-agent systems, smart grids, and optimization. He can be contacted at email: mauricio.mauledoux@unimilitar.edu.co.

Oscar F. Aviles received his B.S. in electronics engineering from Universidad Antonio Nariño, in 1995. He received his PhD degree in Mechanical Engineering (Robotics Devices) in 2008 from Universidade Estadual de Campinas, Brazil. In 1998, he joined the Department of Mechatronics Engineering at Universidad Militar Nueva Granada, Colombia. His current research interests include robotics and biomechatronic. He can be contacted at email: oscar.aviles@unimilitar.edu.co.