Multi-Objective Factors Optimization in Fused Deposition Modelling with Particle Swarm Optimization and Differential Evolution

Mohamed Arezki Mellal (✉ mellal.mohamed@gmail.com)

Center for Advanced Life Cycle Engineering, University of Maryland

Chahinaze Laifaoui

Center for Advanced Life Cycle Engineering, University of Maryland

Fahima Ghezal

Center for Advanced Life Cycle Engineering, University of Maryland

Edward J. Williams

University of Michigan

Research Article

Keywords: rapid prototyping, fused deposition modelling, multi-objective optimization, weighted-sum method, particle swarm optimization, differential evolution

Posted Date: August 10th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-777037/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License. [Read Full License](#)

Version of Record: A version of this preprint was published at International Journal on Interactive Design and Manufacturing (IJIDeM) on March 25th, 2022. See the published version at <https://doi.org/10.1007/s12008-022-00868-3>.

Abstract

The design of any system contemplates the elaboration of a prototype of the entire system or some parts, before the manufacturing phase. Nowadays, rapid prototyping (RP) is widely used by the designers. This paper addresses the multi-objective factors optimization of the fused deposition modelling (FDM) technology. The problem is converted into a single one using the weighted-sum method and then solved by resorting to two nature-inspired computing techniques, namely particle swarm optimization (PSO) and differential evolution (DE). The results obtained are compared.

1. Introduction

For many years, the industry has been continuously evolving due to the increasingly great demand from consumers, and machining processes are widely used, including conventional and nonconventional processes [1, 2]. One of the most important things in the industry is the prototype, which is a miniature or a scaled model of a product to be created. In previous years, the prototype was manually made by molding and it took a long time to manufacture; then with the rapid development of technology, rapid prototyping (RP) can make complex 3D shapes that are made using a Computer Aided Design (CAD) with different processes, to reduce time, minimize cost and achieve complex shapes that cannot be manufactured by machining. Nowadays, various techniques are used, such as 3D printing, or virtual and augmented reality [3, 4]. To obtain a high quality object, it is necessary to optimize the processes. The prototyping processes can be found in several fields, such as medicine, aerospace/aeronautics, and robotics.

Nowadays, the optimization of the prototyping processes remains a challenge for the researchers. Most of the optimization approaches used to solve engineering problems are nature-inspired due to their effectiveness [5]. This paper provides a nonexecutive review of the literature. Comprehensive reviews can found in Refs. [6–14]. Lee et al. [15] used the Taguchi method to optimize the fused deposition modeling (FDM) to produce the acrylonitrile butadiene styrene. Udriou and Nedelcu [16] investigated the inkjet printing and polymer jetting using CATIA software, but the results were not compared to those achieved by alternate methods. Wang et al. [17] used the Taguchi method with the Gray relational analysis to optimize the FDM. The results were compared to the prediction. Sood et al. [18] proposed empirical models for the FDM to improve the tensile, flexural and impact strength responses by considering five process parameters: layer thickness, orientation, raster angle, raster width, and air gap. Variance analysis was used to test the model. Rao and Rai [13] used the teaching-learning-based optimization (TLBO) and the non-dominated sorting TLBO algorithm to solve single and multi-objective prototyping processes, including the model developed by Sood et al. [18]. The Pareto was generated to find the set of optimal solutions. However, selecting one single solution from the Pareto set led to conflicting conclusions. Shirmohammadi et al. [19] implemented the artificial neural network (ANN) with the particle swarm optimization (PSO) to minimize the surface roughness of a 3D printing process. They concluded that the proposed hybrid approach reduced the error.

In this paper, the multi-objective FDM process is optimized with the weighted-sum method and the optimal parameters are provided using the PSO and the differential evolution (DE). The remainder of the paper is organized as follows: Sect. 2 describes the problem. Sections 3 and 4 give the description of the implemented PSO and DE, respectively. Section 5 provides the results with a discussion. Finally, the last section concludes the paper with remarks and suggested directions for future work.

2. Multi-objective Fdm Process

The FDM is a RP process that uses a moving nozzle to extrude a polymeric material fiber [18, 20–23]. Based on empirical models, the multi-objective optimization problem of the FDM is given as follows [13, 18]:

$$\begin{aligned} \text{Maximize } T_s \text{ [MPa]} = & 13.5625 + 0.7156A - 1.3123B + 0.9760C + 0.5183E \\ & + 1.1671A^2 - 1.3014B^2 - 0.4363(A \times C) + 0.4364(A \times D) \\ & - 0.4364(A \times E) + 0.4364(B \times C) + 0.4898(B \times E) \\ & - 0.5389(C \times D) + 0.5389(C \times E) - 0.5389(D \times E) \end{aligned} \quad (1)$$

$$\begin{aligned} \text{Maximize } F_s \text{ [MPa]} = & 29.9178 + 0.8719A - 4.8741B + 2.4251C - 0.9096D \\ & + 1.6626E - 1.7199(A \times C) + 1.7412(A \times D) - 1.1275(A \times E) \\ & + 1.0621(B \times E) + 1.0621(C \times E) - 1.0408(D \times E) \end{aligned} \quad (2)$$

$$\begin{aligned} \text{Maximize } I_s \text{ [MJ/m}^2\text{]} = & 0.401992 + 0.034198A + 0.008356B + 0.013673C \\ & + 0.02138A^2 + 0.008077(B \times D) \end{aligned} \quad (3)$$

Subject to the following design variables:

$$0.127 \text{ mm} \leq A \leq 0.254 \text{ mm} \quad (4)$$

$$0^\circ \leq B \leq 30^\circ \quad (5)$$

$$0^\circ \leq C \leq 60^\circ \quad (6)$$

$$0.4064 \text{ mm} \leq D \leq 0.5064 \text{ mm} \quad (7)$$

$$0 \text{ mm} \leq E \leq 0.008 \text{ mm} \quad (8)$$

where T_s [MPa] is the tensile strength, F_s [MPa] is the flexural strength, I_s [MJ/m²] is the impact strength, A [mm] is the layer thickness, B [degree] is the orientation, C [degree] is the raster angle, D [mm] is the raster width, and E [mm] is the air gap.

In this paper, the multi-objective problem described in Equations (1)-(3) is converted into a single objective problem using the weighted-sum method [24–27] as follows:

$$\text{Maximize } Z = w_1 T_s + w_2 F_s + w_3 I_s \quad (9)$$

where Z is the one-scaled objective function and w_i ($i = 1, 2, 3$; $w_1 + w_2 + w_3 = 1$) are the weight factors for each objective function.

3. Particle Swarm Optimization

The particle swarm optimization (PSO) is a nature-inspired optimization algorithm inspired by the moving mechanism of swarms, such as flocks of birds and schools of fishes. It was initially developed by Kennedy and Eberhart [28]. It is based on the position and the velocity of the particles of the swarm. Details on the PSO can be found in Refs. [28–30]. The pseudo-code of the implemented PSO to solve the above problem is given in Algorithm 1.

Algorithm 1. Pseudo-code of the implemented PSO.

Input the parameters;

Generate random solutions;

While No. of iterations $\leq$ max No. of iterations **Do**

Evaluate the one-scaled objective function and constraint handling;

Move the particles;

Update the position and the velocity;

Find best position and velocity;

End while

Display the results: Z , T_s , F_s , I_s , A , B , C , D , and E .

4. Differential Evolution

The differential evolution (DE) is a nature-inspired optimization algorithm inspired by the population evolution mechanisms. It was developed by Storn and Price [31] and is based on the genetic algorithms [32] but is characterized by more perturbations during iterations. More details on the DE can be found in Refs. [31, 33–35]. The pseudo-code of the implemented DE to solve the above problem is given in Algorithm 2.

Algorithm 2. Pseudo-code of the implemented DE.

Input of the parameters;

Generate random solutions;

While No. of iterations $\leq$ max No. of iterations **Do**

Evaluate the one-scaled objective function and constraint handling;

Perform evolution mechanisms;

Perturbations during iterations;

End while

Display the results: Z , T_s , F_s , I_s , A , B , C , D , and E .

5. Results And Discussion

The problem described has been solved using the above two algorithms and implemented using MATLAB 2017 and run on a PC with the following characteristics: i5 of 2.50 GHz with 4 GB. Each algorithm has been run over ten independent runs. The population size is 20 and the maximum number of iterations is 100. These parameters have been fixed by trial-and-error and based on experience. The weight factors are considered equal, i.e. ($w_1 = w_2 = w_3 = 0.3333$).

Tables 1 and 2 report the results obtained by the PSO and the DE over the ten runs, respectively. The values of the one-scaled objective (Z), the three process performances (T_s , F_s and I_s), design variables, number of function evaluations (NFE), CPU time, and standard deviation (σ) are included.

From Table 1, it can be observed that the best value of Z obtained by the PSO is 98.8849 for all runs, except #3, #6 and #7. The fewer NFE corresponds to #5 (1,220) with 12.23 s of CPU time. The optimal values are $T_s=170.2408$ MPa, $F_s=125.3889$ MPa, $I_s=1.3162$ MJ/m², whereas the decision variables are $A=0.1270$ mm, $B=7.6880^\circ$, $C=60^\circ$, $D=0.4064$ mm, and $E=0.008$ mm. The standard deviation of the ten runs is 2.8E-04.

From Table 2, it can be observed that the best value of Z obtained by the DE is 98.8849 for all runs. The lower NFE corresponds to #9 (440) with 9.89 s of CPU time. The optimal values are $T_s=170.3954$ MPa, $F_s=125.2397$ MPa, $I_s=1.3166$ MJ/m², whereas the decision variables are $A=0.1270$ mm, $B=7.6923^\circ$, $C=60^\circ$, $D=0.4064$ mm, and $E=0.008$ mm. The standard deviation of the ten runs is 0.

Table 3 summarizes the best results obtained by the PSO and the DE. It can be observed that both algorithms obtained the same value of the one-scale objective ($Z=98.8849$). However, the performances of the DE have outperformed those of the PSO in terms of number of function evaluations, CPU time, and standard deviation.

6. Conclusions

The goal of this paper was to investigate the multi-objective fused deposition modelling (FDM) optimization problem which is a rapid prototyping process (RP). The considered objectives were the tensile strength, the flexural strength, and the impact strength. The multi-objective problem was converted to a single one using the weighted-sum method in order to avoid the disadvantages of the Pareto set. The particle swarm optimization (PSO) and differential evolution (DE) were implemented with constraint handling to solve the problem. It was shown that both algorithms provides the same value of the one-scaled objective, but the performances of the PSO were better. Future work will be devoted to the development of a hybrid approach to improve the results and solve other rapid prototyping processes.

Declarations

Authors' contributions Mohamed Arezki Mellal conceived of the study, designed the study, supervised the study, and wrote the manuscript. Chahinaze Laifaouia and Fahima Ghezal investigated the study, programmed the study, and collected the results. Edward J. Williams designed the study and revised the manuscript.

Funding This study has not received funds.

Data availability The data sets supporting the results of this article are included within the article.

References

1. Mellal MA, Williams EJ (2016) Total production time minimization of a multi-pass milling process via cuckoo optimization algorithm. *Int J Adv Manuf Technol*. <https://doi.org/10.1007/s00170-016-8498-3>
2. Mellal MA, Williams EJ (2016) Parameter optimization of advanced machining processes using cuckoo optimization algorithm and hoopoe heuristic. *J Intell Manuf* 27:927–942
3. Meng L, Zhang W, Quan D, et al (2020) From Topology Optimization Design to Additive Manufacturing: Today's Success and Tomorrow's Roadmap. *Arch Comput Methods Eng* 27:805–830. <https://doi.org/10.1007/S11831-019-09331-1>
4. Wang Q, Tan Y, Mei Z (2019) Computational Methods of Acquisition and Processing of 3D Point Cloud Data for Construction Applications. *Arch Comput Methods Eng* 2019 272 27:479–499. <https://doi.org/10.1007/S11831-019-09320-4>
5. Mellal MA, Adjerid S, Williams EJ, Benazzouz D (2012) Optimal replacement policy for obsolete components using cuckoo optimization algorithm based-approach: Dependability context. *J Sci Ind Res (India)* 71:715–721
6. Peltola SM, Melchels FPW, Grijpma DW, Kellomäki M (2008) A review of rapid prototyping techniques for tissue engineering purposes. *Ann Med* 40:268–280. <https://doi.org/10.1080/07853890701881788>
7. Mohamed OA, Masood SH, Bhowmik JL (2015) Optimization of fused deposition modeling process parameters: a review of current research and future prospects. *Adv Manuf* 3:42–53. <https://doi.org/10.1007/s40436-014-0097-7>
8. Kakde NU, Deshbhratar VJ (2016) Rapid prototyping: A review. *Int J Eng Res Gen Sci* 4:114–119
9. Coon C, Pretzel B, Lomax T, Strlič M (2016) Preserving rapid prototypes: A review. *Herit. Sci.* 4:1–16
10. Wiberg A, Persson J, Olvander J (2019) Design for additive manufacturing – a review of available design methods and software. *Rapid Prototyp. J.* 25:1080–1094
11. Plocher J, Panesar A (2019) Review on design and structural optimisation in additive manufacturing: Towards next-generation lightweight structures. *Mater Des* 183:108164. <https://doi.org/10.1016/j.matdes.2019.108164>
12. ZHU J, ZHOU H, WANG C, et al (2021) A review of topology optimization for additive manufacturing: Status and challenges. *Chinese J Aeronaut* 91–110. <https://doi.org/10.1016/j.cja.2020.09.020>
13. Rao RV, Rai DP (2016) Optimization of fused deposition modeling process using teaching-learning-based optimization algorithm. *Eng Sci Technol an Int J*. <https://doi.org/10.1016/j.jestch.2015.09.008>
14. Madheswaran S, Sivakumar K, Prakash E, et al (2021) Applications of additive manufacturing - A review. In: *Advances in Materials Research*. Springer, Singapore, pp 21–27
15. Lee BH, Abdullah J, Khan ZA (2005) Optimization of rapid prototyping parameters for production of flexible ABS object. *J Mater Process Technol*. <https://doi.org/10.1016/j.jmatprotec.2005.02.259>
16. Udriou R, Nedelcu A (2011) Optimization of additive manufacturing processes focused on 3D printing. In: *Rapid Prototyping Technology - Principles and Functional Requirements*. InTech, pp 1–28
17. Wang CC, Lin TW, Hu SS (2007) Optimizing the rapid prototyping process by integrating the Taguchi method with the Gray relational analysis. *Rapid Prototyp J* 13:304–315. <https://doi.org/10.1108/13552540710824814>
18. Sood AK, Ohdar RK, Mahapatra SS (2010) Parametric appraisal of mechanical property of fused deposition modelling processed parts. *Mater Des*. <https://doi.org/10.1016/j.matdes.2009.06.016>
19. Shirmohammadi M, Goushchi SJ, Keshtiban PM (2021) Optimization of 3D printing process parameters to minimize surface roughness with hybrid artificial neural network model and particle swarm algorithm. *Prog Addit Manuf* 1–17. <https://doi.org/10.1007/s40964-021-00166-6>
20. Altuntaş E, Özkan B, Yener G (2017) Porous scaffolds. In: *Nanobiomaterials Science, Development and Evaluation*. Elsevier, pp 27–59
21. Zein I, Huttmacher DW, Tan KC, Teoh SH (2002) Fused deposition modeling of novel scaffold architectures for tissue engineering applications. *Biomaterials* 23:1169–1185. [https://doi.org/10.1016/S0142-9612\(01\)00232-0](https://doi.org/10.1016/S0142-9612(01)00232-0)
22. Sachlos E, Czemuszka JT (2003) Making tissue engineering scaffolds work. Review on the application of solid freeform fabrication technology to the production of tissue engineering scaffolds. *Eur Cells Mater* 5:29–40. <https://doi.org/10.22203/ecm.v005a03>
23. Chim H, Huttmacher DW, Chou AM, et al (2006) A comparative analysis of scaffold material modifications for load-bearing applications in bone tissue engineering. *Int J Oral Maxillofac Surg* 35:928–934. <https://doi.org/10.1016/j.ijom.2006.03.024>

24. Zadeh LA (1963) Optimality and non-scalar-valued performance criteria. IEEE Trans Automat Contr 8:59–60.
<https://doi.org/10.1109/TAC.1963.1105511>
25. Mellal MA, Salhi A, Williams EJ (2020) Accelerated cuckoo optimization algorithm for the multi-objective welding process. J Brazilian Soc Mech Sci Eng 42:6. <https://doi.org/10.1007/s40430-019-2089-9>
26. Mellal MA, Salhi A (2020) Parallel–Series System Optimization by Weighting Sum Methods and Nature-Inspired Computing
27. Mellal MA, Zio E (2019) An adaptive particle swarm optimization method for multi-objective system reliability optimization. Proc Inst Mech Eng Part O J Risk Reliab 1748006X1985281. <https://doi.org/10.1177/1748006X19852814>
28. Kennedy J, Eberhart R (1995) Particle swarm optimization. Neural Networks, 1995 Proceedings, IEEE Int Conf 4:1942–1948 vol.4.
<https://doi.org/10.1109/ICNN.1995.488968>
29. Mellal MA, Williams EJ (2018) A Survey on Ant Colony Optimization, Particle Swarm Optimization, and Cuckoo Algorithms. 37–51.
<https://doi.org/10.4018/978-1-5225-2990-3.ch002>
30. Hamadache I, Mellal MA (2021) Design optimization of car side safety system by particle swarm optimization and grey wolf optimizer. In: Mellal MA, Pecht GM (eds) Nature-Inspired Computing Paradigms in Systems: Reliability, Availability, Maintainability, Safety and Cost (RAMS+C) and Prognostics and Health Management (PHM). Elsevier
31. Storn R, Price K (1995) Differential evolution – A simple and efficient adaptive scheme for global optimization over continuous spaces. Berkeley, CA, USA
32. Holland JH (1975) Adaptation in Natural and Artificial Systems
33. Zio E, Golea LR, Sansavini G (2012) Optimizing protections against cascades in network systems: A modified binary differential evolution algorithm. Reliab Eng Syst Saf 103:72–83. <https://doi.org/10.1016/j.ress.2012.03.007>
34. Zio E, Viadana G (2011) Optimization of the inspection intervals of a safety system in a nuclear power plant by Multi-Objective Differential Evolution (MODE). Reliab Eng Syst Saf 96:1552–1563. <https://doi.org/10.1016/j.ress.2011.06.010>
35. Mellal MA, Zio E (2018) Availability optimization of parallel-series system by evolutionary computation. In: 3rd International Conference on System Reliability and Safety. Barcelona, Spain

Tables

Table 1. Results obtained by the PSO.

Run #	T_s [MPa]	F_s [MPa]	I_s [MJ/m ²]	Z	A [mm]	B [°]	C [°]	D [mm]	E [mm]	NFE	CPU[s]	σ
1	170.2548	125.3745	1.3162	98.8849	0.1270	7.6880	60	0.4064	0.008	1,660	17.28	2.8E-04
2	170.4601	125.1358	1.3167	98.8849	0.1270	7.7105	59.9944	0.4064	0.008	1,960	20.23	
3	170.5864	125.0318	1.3170	98.8845	0.127	7.7331	60	0.4064	0.007	1,980	12.18	
4	170.3061	124.9912	1.3161	98.8849	0.127	7.7191	59.9466	0.4064	0.008	1,840	11.90	
5	170.2408	125.3889	1.3162	98.8849	0.1270	7.6609	60	0.4064	0.008	1,220	12.23	
6	170.2634	125.3570	1.3162	98.884	0.1270	7.6671	59.9989	0.4065	0.008	2,000	12.38	
7	170.3064	125.3283	1.3163	98.8848	0.127	7.6741	60	0.4064	0.008	1,720	11.57	
8	170.3662	125.1776	13165	98.8849	0.1274	7.6934	59.9956	0.4065	0.008	2,000	11.81	
9	170.4134	125.2184	1.3166	98.8849	0.127	7.6967	60	0.4064	0.008	1,980	11.74	
10	170.3698	125.2216	1.3165	98.8849	0.127	7.6931	59.9936	0.4065	0.008	1,680	11.95	

Table 2. Results obtained by the DE.

Table 3. Comparison of PSO vs DE.

Run #	T_s [MPa]	F_s [MPa]	I_s [MJ/m ²]	Z	A [mm]	B [°]	C [°]	D [mm]	E [mm]	NFE	CPU[s]	σ
1	170.3954	125.2397	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	600	15.37	0
2	170.3954	125.2397	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	500	9.90	
3	170.3954	125.2397	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	620	9.78	
4	170.3954	125.2397	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	640	9.38	
5	170.3954	125.2397	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	500	9.52	
6	170.3954	125.2397	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	620	9.43	
7	170.3954	125.2398	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	660	9.83	
8	170.3954	125.2397	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	560	9.18	
9	170.3954	125.2397	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	440	9.89	
10	170.3954	125.2398	1.3166	98.8849	0.1270	7.6923	60	0.4064	0.008	660	9.58	

	Z	NFE	CPU[s]	σ
PSO	98.8849	1,980	12.23	2.8E-04
DE	98.8849	440	9.89	0