

Chapter 5

Lightweight Optimization for Additive Manufacturing Parts Based on Genetic Algorithms, Metamodels and Finite Element Analysis

Rubén Paz, Mario Monzón, Begoña González,
Gabriel Winter and Fernando Ortega

Abstract Additive manufacturing (AM) has become in a competitive method for short series production and high flexibility applications even for functional parts. Few constraints in the manufacturing process involve a great design freedom, allowing minimization of weight by using internal cellular and lattice structures, while minimal mechanical requirements are kept. Weight minimization implies a lower use of material and hence a reduction in manufacturing time, leading to a cost reduction. However, design optimization requires a greater effort in the design process, which also results in more costs. In order to reduce the design process, an optimization method based on genetic algorithms (GAs) and computer aided design/finite element method (CAD/FEM) simulations is proposed to optimize the cellular structure design and minimize the weight for AM parts. New optimization strategies based on GAs combined with surrogate models are evaluated and compared to reduce as much as possible the number of FEM simulations.

R. Paz (✉) · M. Monzón · F. Ortega
Departamento de Ingeniería Mecánica, Universidad de Las Palmas de Gran Canaria,
35017 Las Palmas G.C, Spain
e-mail: rpaz@lfi.gi.ulpgc.es

M. Monzón
e-mail: mmonzon@dim.ulpgc.es

F. Ortega
e-mail: fortega@dim.ulpgc.es

B. González · G. Winter
Institute of Intelligent Systems and Numerical Applications in Engineering—SIANI,
Evolutionary Computation and Applications (CEANI),
Universidad de Las Palmas de Gran Canaria, 35017 Las Palmas G.C, Spain
e-mail: bgonzalez@iusiani.ulpgc.es

G. Winter
e-mail: gabw@step.es

Keywords Lightweight optimization · Additive manufacturing · Genetic algorithms · Metamodels

5.1 Introduction and Objectives

The continuous evolution of AM technologies in terms of materials, reliability and reproducibility of the processes, as well as cost reduction compared with conventional manufacturing techniques have led to an increasing use of these technologies in the industry. The low manufacturing constraints associated to these processes involve an enormous design freedom ideal to manufacture complex parts without any cost increase.

Moreover, the possibilities around the CAD/FEM software are well known. FEM numerical simulations allow the determination of the mechanical behavior of any part, being a fundamental tool in the design process. Combining the potential of AM technologies and CAD/FEM tools is possible to reduce the part weight by introducing cellular structures repeated inside the part (without changing the previous external design) [1]. Cellular structures can be generated and parameterized in a CAD model without an excessive effort, especially if the cells are defined with a repeated pattern. FEM simulations of any new design with cellular structures allow knowing its mechanical properties, information that is essential for the design process. Weight minimization can be achieved by optimizing the cell pattern dimensions with an optimization method. In fact, results of finite element analysis (FEA) can be employed to the evaluation of the fitness function in an optimal searching with GAs [3]. Finally, the best design can be manufactured by AM technologies despite the complex internal cellular structure.

Weight minimization not only means a greater efficiency in multiple applications, but also significant reduction of manufacturing costs, either material savings or manufacturing time. However, the extra time required for the design optimization also entails a cost increase, which means that the optimization time must be minimized in order to obtain a more competitive product.

For these reasons, weight minimization will be made through repeated cell geometries (from a pattern) inside the part, which implies less variability of individuals and a smaller number of design variables (less than 7). Although this simplification reduces the searching space and probably the quality of the optimal individual, greatly facilitates the CAD modeling and optimization tasks, significantly reducing the design costs.

Moreover, the evaluation of the fitness function for each individual generated during the GA evolution requires FEM simulations. This would involve an excessive computational time [3]. Therefore, the use of surrogate models to estimate the FEM results without doing the simulations is proposed, reducing the number of the computationally expensive analyses as much as possible [5]. The aim of this

approach is to establish a simple methodology that can be used by any AM user through commercial CAD/FEM software. These technologies are becoming more and more affordable due to cost decreasing associated to the patent expiration. Thus, SMEs or even particular users will be able to buy AM machines and manufacture their own parts, taking advantage of the optimization strategies developed in this proposal through commercial CAD/FEM software easily accessible.

5.2 Main Program Structure

To create a surrogate model it is necessary a previous information of the system behavior. This information is achieved through an initial design of experiments (DOE), where a set of designs are simulated by FEM.

Once the surrogate model is defined, a GA is applied to search the optimal design by evaluating the fitness function of each individual through the surrogate model estimations. Although different versions of the program were tested, the general approach is to refine the metamodel by simulating new designs strategically located in interesting regions, including the results into the database to upload the metamodel. Once the surrogate model guarantees a certain level of accuracy in the estimations, the optimal design is searched again using GAs and metamodel to evaluate the fitness function, reducing as much as possible the number of FEM simulations.

The design variables are related to the dimensions of the pattern cell geometry repeated inside the part, existing in this case a monotonic relation with the system responses. An increase in a variable associated with the hollow cell size always involves a mass reduction and a worse mechanical behavior, adversely affecting the problem optimization constraints (displacements or stresses). This particular relation implies that the optimal design will be always in the border between the feasible and unfeasible regions, so that the optimum will have at least one constraint very close to its limit value.

Given this fact, the addition of new points in the DOE phase or metamodel refinement stage is carried out trying to increase the sampling density in areas close to this border (feasible/unfeasible), which means simulating designs in interesting zones. This strategy of DOE and surrogate model refinement also implies a better fitting in the feasible/unfeasible border than in other regions of the searching space. Refinement strategies usually add new sample points where the lowest accuracy of the metamodel are estimated. However, in this proposal the accuracy of the metamodel is only relevant in the feasible/unfeasible zones. In other regions the surrogate model has a lower precision, but enough to estimate the results without affecting the convergence of the GAs. This refinement method requires a lower sampling, allocating the new points in areas near to the optimum and consequently maximizing the sampling effort.

5.3 Comparison Between Different Metamodels

First a comparison between different surrogated models was carried out to determine the most appropriated metamodels in terms of estimation error for this application. The evaluated metamodels were:

- Inverse distance interpolation: Four different configurations of this interpolation method were evaluated. The first one was implemented using an exponent of 2 in the inverse distance calculation and involving all the available data (IDI2). This same method (with exponent 2 in the inverse distance calculation) was applied again but involving only the 6 nearest data to the point to be estimated (IDI2 6p). After that, the inverse distance exponent was increased to 3, considering all available data in the estimations (IDI3). Finally, a fourth configuration was carried out by using again exponent 3 in the inverse distance calculation but taking into account only the 6 nearest data to the point to be estimated (IDI3 6p). Some authors have observed better results for low sampling problems with this method than with other more complex ones [7].
- Spline interpolation (SI). The main advantage of this method is that it allows the interpolation of values that are below the minimum or above the maximum of the available data, while other methods cannot [2].
- Least square fitting: Two different configurations of the least square fitting were evaluated. The first configuration was developed by fitting the coefficients of a two-order polynomial to the available data (LSF2). The second configuration was carried out in a similar way but using a three-order polynomial to be fitted to the available data (LSF3). The use of polynomial equations in fitting problems with unknown response is a common practice, although the most usual practice is to employ one- or two-order equations [6].
- Linear interpolation based on Delaunay triangulation (LIDT). This method partitions the space into discrete simplex (n-dimensional) following the Delaunay triangulation (dual to the Voronoi diagram or Thiessen polygons). Given a set of points (P) in the n-dimensional space, the Delaunay triangulation is a triangulation such that no point in P is inside the circumhypersphere of any simplex. This method maximizes the minimum angle of all the simplexes. Once the space is discretized according to Delaunay triangulation, the method identifies the simplex of the point to be evaluated and finally a linear interpolation of the vertex values is applied (by a weighted sum of the vertex values, being the weights the barycentric coordinates). The main disadvantage of this interpolation method (also known as Triangulated Irregular Network, TIN) is that the domain is limited to the convex envelope of the data and the resulting surface is not smooth [2].
- Nearest neighbor interpolation (NNI). The NNI selects the value of the nearest point. This is equivalent to the Voronoi diagram, which means that the interpolation values will be the same in each one of the tessellation cells. This method is less accurate but quite simple.

A problem of four design variables was employed in order to compare the accuracy or estimation error among the different surrogated models mentioned above.

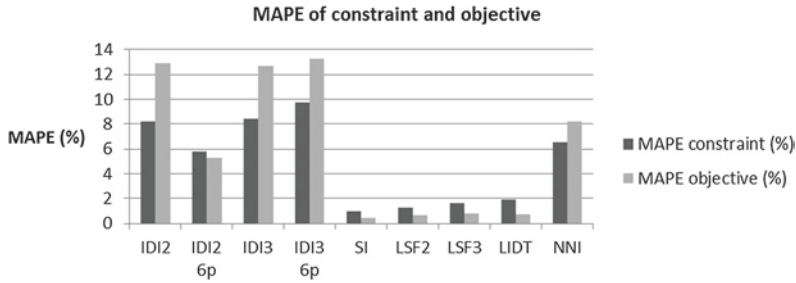


Fig. 5.1 MAPE for evaluated metamodels

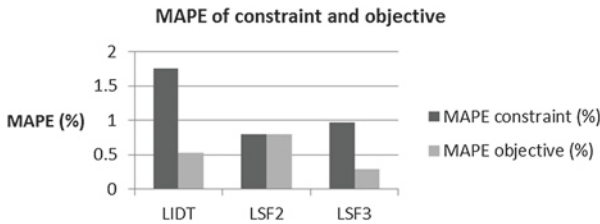


Fig. 5.2 MAPE for least square fitting and linear interpolation metamodels

Eighty one individuals related to a 3-level full factorial DOE were evaluated by FEM simulations. The different types of metamodels were constructed with the data obtained in the previous simulations. After that, 10 random points of the domain were evaluated by FEM and were also estimated by the surrogate models in order to evaluate the estimation error (difference between the estimated value and the FEM result in absolute value). Figure 5.1 shows the mean absolute percentage error (MAPE) of these 10 points for the different metamodels and for the 2 responses of the problem.

Best results were obtained for SI, LSF2/LSF3 and LIDT. Although SI provides more accurate results for both responses, the data distribution required to construct the spline must be in a grid, which complicates the refinement tasks and implies an enormous sampling intensity even using T-splines with the “quadtree” method [4].

Another similar study was made comparing only the LSF2, LSF3 and LIDT. In this case 33 different designs were evaluated by FEM. The allocation of these points was defined according to the first optimization strategy developed in Sects. 5.5 and 5.5.1. The first 17 points correspond to a 2-level full factorial DOE and central point, and the 16 remaining points correspond to an iteration of the border (feasible/unfeasible) approximation. The metamodels were elaborated from the results of these 33 sampling points. Then 16 new points associated with a new iteration of the border approximation were evaluated either by FEM or by the predictions of the surrogate models. Figure 5.2 shows the MAPE obtained for LIDT, LSF2 and LSF3, for both constraint and objective responses.

Although the results of LSF2 and LSF3 are even better than those of LIDT, this last metamodel was chosen for this proposal because it is an interpolation method, which means exact predictions on the data points and ensures greater accuracy than the least squares fitting when the sampling is intensified in an area. The least square fitting does not estimate exact results in the data points and its potential is limited by the shape of the equation to be fitted, which could imply a high distortion and error in some areas when the sampling density is increased in a specific zone. Furthermore, the refinement strategy discussed above will only work correctly with a surrogate model which can improve its accuracy as new points are added.

5.4 Genetic Algorithm

A GA with different configurations was implemented to solve a known problem with four variables (real numbers), one constraint and one objective to be minimized. This reference problem was employed to validate the different programs developed in this paper without doing the FEM simulations. The fitness value of the theoretical optimal design is $F = 1600.809$. The number of individuals of the population was fixed in 100, with a tournament selection of 2 individuals, arithmetic crossover and application of elitism. The parameters of the different GA configurations were as follows:

- Penalty amplification factor (AF): individuals who did not satisfy any constraint of the optimization problem were penalized with a certain penalty which was amplified by a factor defined as a fixed value or as a value that grew exponentially with the number of generations. This latter option seeks to assign a greater freedom in the first iterations of the GA and become more restrictive as the GA evolves.
- Type of penalty: individuals who did not satisfy any constraint of the optimization problem were penalized with an error value which was obtained from the squared error (SE) or absolute error (AE), always amplified by the penalty amplification factor mentioned above. The total penalty for each individual was determined as the sum of the penalties associated with each of the non-satisfied restrictions.
- Total number of generations evaluated: 50 or 100.
- Cross probability: 50 or 80 %.
- Mutation probability: mutation probability was defined as a fixed value or a variable value that increases linearly with the number of generations. This latter option seeks to provide a greater localized variability as the GA evolves, in order to avoid convergence problems (convergence to local optima instead of global optima). The more evolved the population is, the higher the mutation probability becomes, which reduces the problems of stagnation in a local optimum.

Table 5.1 Different configurations tested

Geometry parameters	Config. 1	Config. 2	Config. 3	Config. 4	Config. 5	Config. 6
Amplification factor	10^{99}	10^{gen}	10^{gen}	10^{gen}	10^{99}	10^{99}
Penalty	SE·AF	SE·AF	SE·AF	AE·AF	SE·AF	SE·AF
Number of generations	50	50	100	50	50	50
Cross probability (%)	80	80	80	80	50	50
Mutation probability (%)	10	10	10	0–60	60	80
Mutation amplitude (%)	10	10	10	0–50	50	10
Fitness average (F)	1615.3	1615.0	1606.3	1601.7	1601.9	1601.8

- Mutation amplitude: individuals were randomly mutated with a maximum amplitude value defined by a fixed value or variable with the number of generations. This latter option seeks to provide a more intensive mutation as the GA evolves in order to improve the convergence to the optimal.

Table 5.1 shows a summary of the six configurations tested and an average value of the optimal fitness function for 10 different runs. The best results were obtained for configurations 4, 5 and 6, with fitness values very close to the theoretical optimal value. This means that the GA can converge to the optimal with different configurations, which demonstrates its robustness and flexibility. Configurations 5 and 6 were chosen to be implemented in the different developed programs, while configuration 4 was rejected due to be a more complex option.

5.5 Optimization Programs Developed

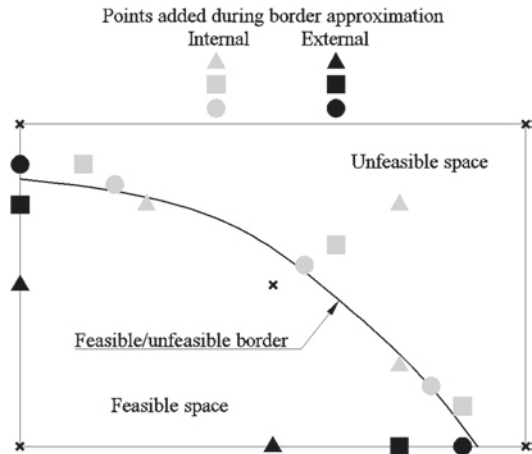
Different optimization strategies were developed and tested with the previous known problem (without FEM simulations). The last 2 versions were also tested with a case study with FEM analysis.

5.5.1 Version 1

The first version consist of a 2-level full factorial DOE and central point, followed by a phase of addition of points near to the feasible/unfeasible border and finally a GA.

LIDT can only be applied inside the convex hull of the data. For this reason, it is important that the initial DOE allows the creation of a convex hull that covers the entire domain, in order to apply LIDT throughout the searching space. A 2-level full factorial DOE was chosen as the best option to achieve the desired convex hull with the minimal number of points (which is equivalent to evaluating

Fig. 5.3 Points added during the initial DOE and border approximation in a 2D problem



all vertices of the domain). Apart from these points, the central point of the domain was also simulated in this initial stage of DOE (black crosses in Fig. 5.3).

The phase of approximation to the feasible/unfeasible border was divided into 2 parts: (a) an “internal” approximation by adding new points in the middle between the central point and the corners when one of them is in the feasible space and the other does not (or vice versa); (b) an approximation “along the edges of the domain”, adding new points in the middle between adjacent corners when one of them is in a feasible area and the other does not (square, triangle and circle points in Fig. 5.3). This new points will be closer to the feasiably/unfeasible border and consequently closer to the optimal design. This phase is repeated in a loop until the mean absolute deviation of the points added in the last iteration is less than the maximum deviation assigned by the user to each response. Triangles, squares and circles of Fig. 5.3 represent the points added in 3 iterations respectively, both internal (in grey) and external (in black) approximation.

Finally, a GA based on configuration 5 is applied. In this case, the fitness value is evaluated by LIDT through the available data of previous simulations. In addition, the best individual of each generation is simulated and then the information obtained is added to the available data in order to refine the metamodel, thus getting a more accurate metamodel as the GA evolves. Once the GA ends, the best simulated design is chosen.

This version was executed 10 times with the reference problem (5 tests with 100 generations and 5 with 500). The average value of the optimal fitness functions was 1659.267 and 1660.593 respectively, so no improvements were observed by increasing the number of generations. The total average value of fitness function was 1659.93 (with an average of 80 points evaluated), which differs significantly from the fitness value of the theoretical optimum ($F = 1600.809$). After several tests, it was observed that the GA does not converge to the theoretical optimum because of the lack of accuracy of the metamodel. Hence, the simulation of

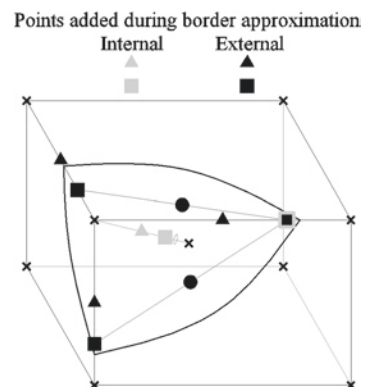
the best individual of each generation does not significantly improve the fit of the metamodel. So, new points must be added before applying the GA if the results are intended to be improved.

5.5.2 Version 2

In order to improve the results, new middle points were added between the point with minimum mass found in the last iteration of the border approximation along the edges and the remaining points associated with adjacent corners of the feasible/unfeasible border. This approach was implemented in version 2. Figure 5.4 shows this new strategy in a 3D problem. The square black point outlined in grey represents the point with lower mass among the square black points added in the last iteration of the border approximation along the edges. This point is combined with the remaining adjacent square black points to obtain the two middle points represented as black circles in Fig. 5.4.

Additionally, the phase of border approximation (internally and along the edges) was carried out by linear interpolation to improve the convergence to the border, identifying the two closest points to the border (each in the opposite feasible/unfeasible space) and allocating the new point on the border line estimated by linear interpolation of the 2 selected data for each constraint of the problem. Finally, the proposed point that is closer to the feasible zone (corresponding to the most restrictive constraint) is simulated. For example, in the case of having 2 constraints involved in a border approximation along one edge (see Fig. 5.5), each involved constraint will lead to a proposed point (square and triangle points). These points are obtained by linear interpolation of the constraint values associated with the 2 closest data on this edge (one in the feasible zone of the specific constraint and another one in the unfeasible zone). Therefore, only one of the 2 proposed points must be chosen. The closest to the feasible vertex of the edge will

Fig. 5.4 New middle points (black circles) added in a 3D problem



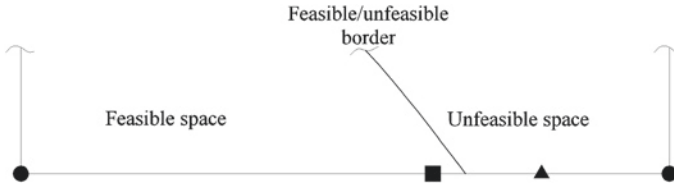


Fig. 5.5 Border approximation phase through different constraints and selection of the proposed point which is closer to the feasible zone

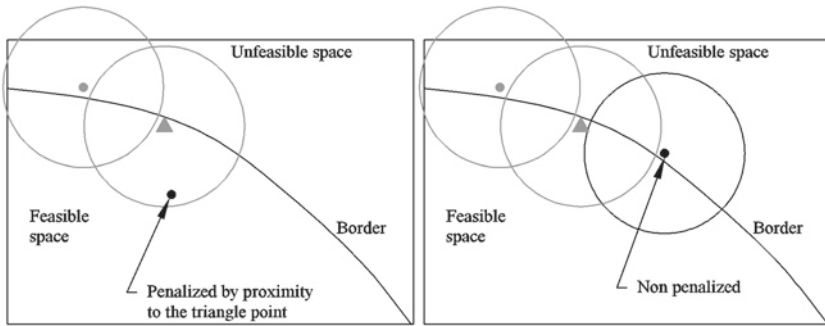
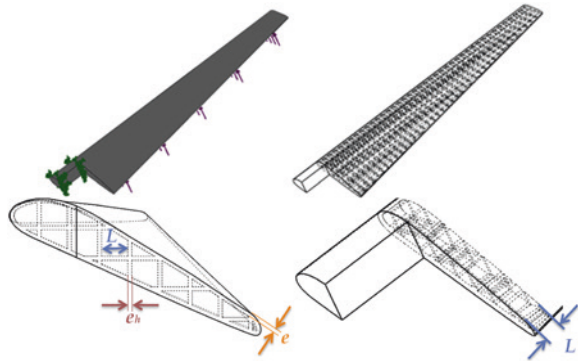


Fig. 5.6 Proximity penalty strategy during the border approximation phase by GAs

be selected (in this case the square point). This step is repeated while the MAPE of the critical constraint value of the different points added in the last iteration (compared to the limit value) is greater than 1 %.

Afterwards a border approximation phase by GAs (configuration 5 with 100 generations) was implemented, using LIDT to evaluate the fitness function. The best individual achieved by the GA is analyzed by FEM and added to the database to upload the metamodel. The GA is executed again, but penalizing the individual that is near to the points added previously in this phase. This strategy converges to a different optimum in each successive GA execution, which involves adding different new points along the feasible/unfeasible border, exploring interesting zones. For example, the circle black point (Fig. 5.6, left image) would be penalized by proximity to the triangle grey point (added in the previous execution of the GA). Hence, the GA evolves towards a point out of the proximity penalty radius of the points added in this stage of the program. Once the circle black point (Fig. 5.6, right image) is evaluated, the GA is executed again but also penalizing the proximity to this new point. This step is repeated until at least “n” points ($n = \text{number of design variables}$) have been added in this phase. After that, the MAPE of the last added point (response estimations compared to simulations) is evaluated. If the MAPE is bigger than 1 %, the metamodel is uploaded with this last point and this GA is executed again applying proximity penalty. And so on until the MAPE value is less than 1 %.

Fig. 5.7 Case study geometry and design variables



Subsequently a final GA (configuration 6 with 200 generations) is run, using LIDT to calculate the fitness function value. The best individual is simulated by FEM. If it is in the feasible zone, is the optimum, otherwise the results are added to the database and the metamodel is uploaded to execute again this final GA. And so on until reaching a feasible optimum.

In 10 different runs of this program version with the reference problem, the average value of the optimal fitness function was $F = 1603.715$, very close to the fitness value of the theoretical optimum (1600.809), with an average of 62 evaluated designs.

A case study with FEM simulations (see Fig. 5.7) in which it is pretended to minimize the weight of a blade for wind power micro-turbine lightened by cellular structures (3 design variables) keeping the maximum deflection under 15 mm (constraint) was also solved. The 3 design variables (see Fig. 5.7) were the length of the sides of the cubic hollows (“L”, varying between 20 and 60 mm), the external thickness (“e”, varying between 3 and 8 mm) and the thickness between the cubic hollows (“e_h” varying between 3 and 8 mm).

The optimization problem can be represented as follows:

$$\begin{aligned}
 & \text{Minimize } \text{mass}(L, e, e_h) \\
 & \text{Subject to } \text{max.deflection} \leq 15 \\
 & \quad 20 \leq L \leq 60 \\
 & \quad 3 \leq e \leq 8 \\
 & \quad 3 \leq e_h \leq 8 \\
 & \text{Individual } (L, e, e_h)
 \end{aligned} \tag{1}$$

Figure 5.8 shows the responses for each one of the 40 designs evaluated during the program evolution (maximum deflection and weight). The weight values were divided by the weight of the optimal design obtained (1632.55 g), while the deflection values were divided by the maximum permitted deflection (15 mm), representing then the relative values of both responses in the same graphic. It can be observed how the relative deflection tends to 1, which means that the program

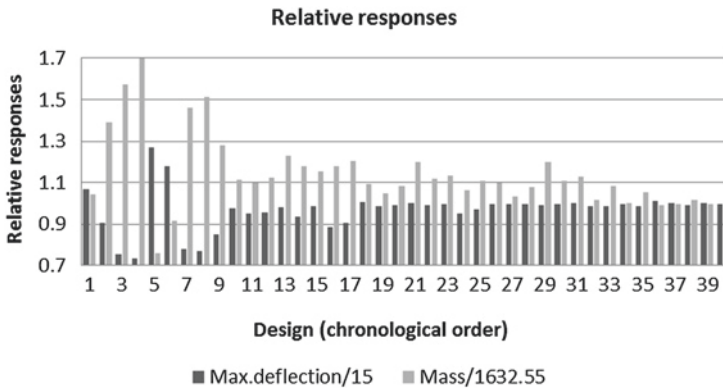


Fig. 5.8 Relative responses of the designs evaluated during the optimization process

evolves to designs with a maximum deflection close to 15 mm in order to minimize the weight as much as possible. Therefore, as it was expected, best design is near to the feasible/unfeasible border.

The optimal design obtained after 40 FEM simulations has a mass of 1632.55 g and 14.992 mm of maximum deflection, being its design variables “ $L = 39.875$ mm”, “ $e = 4.091$ mm” and “ $e_h = 3$ mm”. This same problem was also solved by an optimization method based on Box-Behnken DOE and optimal estimation by response surface method (BBRS), an optimization strategy available in the commercial software of design and FEM simulations, SolidWorks. Response Surface Methods (RSMs) are considered a very effective approach for optimization problems with a small number of design variables, which is ideal for this application. The BBRS method achieves an optimal of 1690.07 g (“ $L = 47.531$ mm”, “ $e = 4.496$ mm” and “ $e_h = 3.193$ mm”) with only 14 simulations. The proposed methodology reaches an optimum 3.52 % better but requires quite more simulations. For this reason, a new version was developed in order to reduce the number of FEM simulations required during the evolution of the optimization algorithm.

5.5.3 Version 3

The last two phases of the program (based on GAs) were tested by excluding different data set in order to evaluate the convergence of the program to the optimum without these points. These tests were carried out solving the reference problem without FEM analysis in order to accelerate the process. The conclusions obtained in this analysis are listed below:

- The points added during the internal feasible/unfeasible border approximation were deleted and the algorithm evolved to practically the same optimal solution, which means that the points added during the internal border approximation have no effect on the quality of the optimum. For this reason, this step was excluded.
- The phase of border approximation along the edges was carried out only edge by edge, achieving a deviation from the real feasible/unfeasible border less than 1 % at each affected edge. This new strategy showed a significant improvement in the solution because it helps to correctly select the best corner of the feasible/unfeasible border for the next phase of the code. Furthermore, this idea has the advantage of varying the number of iterations depending on the design variable that is being changed during the border approximation. This means that the border approximation through one of the design variables may need 5 iterations to achieve a deviation less than 1 % at the most restrictive constraint, while the border approximation along another edge may need just 2 iterations. Hence, the control statement in the new program (version 3) is just the difference (absolute value) between the critical value of the most restrictive constraint and the value obtained for this same constraint in the simulation of the last point added during the border approximation, thus controlling the deviation in the associated edge. However, in the previous version, the border approximation was carried out by adding a new point in each affected edges, repeating this process if the MAPE of the points simulated in this iteration was greater than 1 %. So, the number of points added during the border approximation was the same in all the edges affected and thus some points were probably closer to the border than others, increasing the risk of error in the next phase of the program, where it must be selected the best corner (best point added in the last iteration) of the border between feasible and unfeasible regions.
- Despite the proposal has been developed for a small number of design variables (less than 7), the addition of new middle points between the best border corner and the remaining adjacent corners involves incorporating a lot of points, growing exponentially as the number of design variables increases. However, after some tests it was observed that the method also converged to the theoretical optimum just combining the best border corner with the “n-1” best remaining adjacent corners. For this reason, version 3 was implemented with this new strategy, which means to combine only the best border corner with the “n-1” best remaining adjacent corners instead of considering all possible combinations, hence reducing the sampling intensity.

The new version 3 was programmed and then it was executed 10 times for solving the reference problem (without FEM analysis). The average value of the optimal fitness function was $F = 1606.050$, with an average of 44 sampling points. The new version converges to a solution 0.15 % worse than the previous version, but requires only 44 instead of 62 simulations, reducing in approximately 29 % the CPU time if a linear relation between CPU time and number of designs evaluated by FEM is assumed.

Table 5.2 Some of the designs evaluated during the evolution of version 3 (with FEA)

Point	L (mm)	e (mm)	e _h (mm)	Maximum deflection (mm)	Mass (g)
1	20	3	3	16.061	1705.09
8	60	8	8	11.543	2470.68
9	40	5.5	5.5	12.741	2087.14
11	20	3	4.7	14.905	1948.70
13	20	3.562	3	14.975	1814.40
15	35.157	3	8	15.015	1847.10
18	60	4.577	3	14.851	1643.70
22	60	4.115	8	15.045	1741.10
23	60	4.346	5.5	14.987	1694.90
24	40	4.07	3	15.04	1626.80
25	60	4.521	3	15.07	1630.30
27	31.33	3.858	3	14.943	1668.40
28	56.705	4.479	3	15.043	1631.60
29	60	4.54	3	14.975	1634.85

Applying this new version in the previous case study (with FEM analysis), an optimal design of 1634.85 g and only 29 sampling points was found (“L = 60.000 mm”, “e = 4.540 mm” and “e_h = 3.000 mm”). This optimal design increased the mass by 0.14 % compared to the optimum obtained with version 2, but the number of evaluated designs was reduced from 40 to 29 (approximately 27.5 % of CPU time reduction). Compared with the result of BBRS method, this program improves the optimal 3.38 % but requires more sampling points (29 vs. 14). However, it ensures the convergence to the theoretical optimum due to the refinement loops, while the BBRS method does not guarantee the convergence to a feasible design and its refinement is quite limited by the equation shape to be fitted. In addition, it should be noted that the sampling point number 18, which is added during the border approximation along the edges, improves the optimum obtained by BBRS method with only 4 more simulations (see Table 5.2).

Table 5.2 shows most of the designs evaluated during the optimization process. Points from 1 to 9 correspond to the initial DOE (2-level full factorial DOE and central point). Points 10–11, 12–13, 14–15, 16–18 and 19–22 are added during the border approximation (5 different edges of the domain). Points 23 and 24 are associated to the middle points added between the best corner of the border (point 18) and the 2 best remaining adjacent corners (points 22 and 13). Points 25–27 are added during the phase of exploration along the feasible/unfeasible border by GAs with proximity penalty. Finally, points 28 and 29 are added in 2 different executions of the final GA.

5.6 Conclusions

A new lightweight optimization method for cellular structures in AM has been presented, based on a 2-level full factorial DOE and central point, border approximation along the edges, addition of new middle points between the best border corner and the best “n-1” adjacent remaining corners, addition of new points along the feasible/unfeasible border using GAs with proximity penalty and LIDT metamodel, and a final optimal searching through a GA also combined with LIDT metamodel.

The border approximation phase along the edges allows achieving good designs with a low sampling effort in the case of a small number of design variables. Moreover, in many cases, the optimum is on the boundary of the domain. For this reason, the border approximation phase along the edges is a good and simple strategy for problems with a small number of design variables.

The proximity penalty in the GA allows the addition of new points along the feasible/unfeasible border (interesting zones). These points improve the fitting of the surrogate model in areas where the optimal will be found. Hence, in the next executions of the GAs, the algorithm leads to solutions closer to the theoretical optimum thanks to the refinement achieved during this stage of the program.

Finally, it should be also noted that the linear interpolation metamodel drastically reduced the FEM simulations, obtaining a methodology that guarantees convergence to the optimal design with a low sampling density.

Although this proposal achieves good results in lightweight optimization of cellular structures for Additive Manufacturing parts, further research must be conducted in the future to further reduce the number of FEA and consequently allow design cost savings. In addition, new strategies must be developed in order to apply this concept in problems with a larger number of design variables.

References

1. Chu C, Graf G, Rosen DW (2008) Design for additive manufacturing of cellular structures. *Comput Aided Des Appl* 5(5): 686–696. doi:[10.3722/cadaps.2008.686-696](https://doi.org/10.3722/cadaps.2008.686-696)
2. Dressler M (2009) Art of surface interpolation. In: PhD thesis: Technical University of Liberec, Faculty of Mechatronics and Interdisciplinary Engineering Studies, Kunštát, 2009. <http://m.dressler.sweb.cz/AOSIM.pdf>
3. Gätzi R, Uebersax M, König O (2000) Structural optimization tool using genetic algorithms and ansys. In: Proceedings 18 CAD-FEM User's Meeting, Internationale FEM-Technologietage, Graf-Zeppelin-Haus, Friedrichshafen, 2000. http://www.powerpac.ethz.ch/pub-pat/optimierung_mittels_genetischen_algorithmen.pdf
4. González-López JI (2012) Modelización de sólidos para el análisis isogeométrico. Trabajo Fin de Máster. Máster Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería, Universidad de Las Palmas de Gran Canaria, 2012. <http://acceda.ulpgc.es/handle/10553/7600>

5. Lee JH, Hwang SC, Park JH, Lee KH (2010) Structural design examples using metamodel-based approximation model. In: Proceedings of the 9th WSEAS International Conference on Applied Computer and Applied Computational Science, World Scientific and Engineering Academy and Society (WSEAS), pp 153–156. <http://www.wseas.us/e-library/conferences/2010/Hangzhou/ACACOS/ACACOS-26.pdf>
6. Melcón CDF, Piñeiro-Barcia M (2004) Superficies de Respuesta Métodos y Diseños, Métodos y diseños. In: Diseño de Experimentos. Departamento de Estadística e Investigación Operativa y Didáctica de la Matemática, 2004. http://www.fcb.unl.edu.ar/laboratorios/ladaq/curso_TopQuim_2013/Bibliografia%20RSM/superficie%20de%20respuesta%201.pdf
7. Villatoro M, Henríquez C, Sancho F (2008) Comparación de los interpoladores IDW y Kriging en la variación espacial de pH, Ca, CICE y P del suelo. *Agronomía Costarricense* 32(1): 95–105. <http://estudiosterritoriales.org/resumen.oa?id=43632109>