Estimation of Distribution Algorithm (EDA) for design and optimization of arch structures

Jonathan Melchiorre[1*[0000-0002-8721-8365]], Amedeo Manuello[1[0000-0003-1474-0176]], Marco Martino Rosso[1[0000-0002-9098-4132]], Vincenzo Morganti[1], Gabriele Rosi[1], and Giuseppe Carlo Marano[1,2[0000-0002-8748-0312]]

[1] Politecnico di Torino, DISEG, Department of Structural, Geotechnical and Building Engineering, Corso Duca Degli Abruzzi, 24, Turin 10128, Italy.
jonathan.melchiorre@polito.it

[2] College of Civil Engineering, Fuzhou University, Fuzhou 350108, China.

Abstract. The architecture, engineering, and construction sector has stressed the concepts of efficiency and sustainability because of the industry's high resource consumption. Efficiency is a major criterion for building large-scale structures in order to meet performance requirements while using the least amount of structural material. With the emergence and development of computer tools, mathematical computation-based structural optimization has become a popular way of designing sustainable and efficient buildings. Classical and traditional procedures might not always produce an optimal solution effectively because optimization problems are often non-linear, discontinuous and complex. As a result of their versatility, metaheuristic algorithms have grown in popularity, notably in the engineering profession. The effectiveness of different metaheuristic algorithms, including the Genetic Algorithm (GA), Simulated Annealing (SA), and Estimation of Distribution Algorithm (EDA), is investigated in this work. As a case study, a parametric arch structure is utilized. Numerical results are achieved by minimizing structural weight while checking that the Von Mises stress is lower than the yielding limit all along the arch. Three single-objective algorithms are used in the study, and their performance is objectively evaluated. The parametric modeling, structural analysis, and optimization phases are conducted entirely in the virtual environment offered by the software packages Rhino, Grasshopper, Karamba, and Galapagos. These interoperable software programs were chosen because of their flexibility, customizability, and widespread usage in engineering and architectural offices, making the techniques and findings of this study valuable to design practitioners.

Keywords: Arch structures · Parametric Design · Structural Optimization · Grasshopper · Meta-heuristic algorithms.

1 Introduction

In recent decades, the significance of sustainability and efficiency has grown within the architecture, engineering, and construction (AEC) industry. Given the massive material consumption connected with the construction industry, efficiency is paramount in designing large-scale structures [1]. The structural design can be considered successful when the target performance is attained with the least amount of structural material. Since the introduction and development of computational tools, structural optimization has become one of the most widely utilized methodologies for designing sustainable and efficient structures [2]. Due to the complexity of optimization problems, conventional and traditional methods may not always efficiently produce a globally optimal solution. This is especially true for structural typologies where the shape is closely intertwined with the distribution of stress, as in the case of arch structures [3, 4]. In the past, numerous techniques for form-finding [5] and structural optimization have been developed to determine the optimal shape of such structures [6]. In structural optimization problems, the objective functions are frequently non-differentiable and have a discontinuous search space. Using gradient-based or quasi-newton approaches is frequently not an option. In this context, metaheuristic algorithms are a valuable tool for solving constrained optimization problems because they do not require gradient information. Among the metaheuristic optimization methods, the estimation of distribution algorithm (EDA) [7]is a useful tool for dealing with structural engineering problems [8]. In this work, a parametric population-based EDA is presented. The algorithm is based on the iterative update of a parametric Gaussian Mixture Model (GMM). A self-made code has been developed in Matlab. Then the method was implemented as a tool in the Grasshopper environment to be used by practitioners. Grasshopper is a visual-coding software that runs in Rhinoceros 3D environment. It is widely used for the parametric design of structures. In Grasshopper, a parametric design of arch structures has been defined and optimized by the use of EDA. Cases with 2, 3, 4, and 5 parameters were studied based on the number of design variables considered. The objective function of the optimization problem has been set in order to minimize the structural material consumption while matching the design requirements. The numerical results obtained by the EDA have been compared with those obtained by two optimization algorithms already implemented in Grasshopper. The comparison between EDA , Genetic Algorithm (GA) [9] and Simulated Annealing (SA)) [10] was performed in order to investigate the effectiveness of the metaheuristics algorithms. Finally, a sensitivity analysis of the EDA algorithm was conducted for different GMM parameters. Comparative graphs are reported to assess the performance of the various tested algorithms.

2 Parametric population-based EDA

The current study introduces a novel implementation of the Estimation of Distribution Algorithm (EDA). The proposed EDA version is a parametric, population-based, metaheruistic algorithm. The proposed methodology is intended both for constrained and unconstrined optimization problems. The EDA starts by generating a population of m individuals by the Latin Hyper-cube Sampling (LHS). Each individual of the initial population is evaluated both in terms of objective function and constraint violation. In the unfortunate case in which the entire population is unfeasible the objective function is substitued by the new goal of minimizing the constraint violation. This approach aims to find a feasible reagion. The search space is a space of n_{dim} dimensions where n_{dim} is the number of design variables. Considering x_i the generical design variable, the design space can be defined as Cartesian product $\Omega = [x_1^{lb}; x_1^{ub}] \times ... \times [x_{n_{dim}}^{lb}; x_{n_{dim}}^{ub}]$.

The information obtained on the objective function and constraints violation are used to define a probabilistic model. The main idea is to use the model to sample the next generation of populations. The model is defined in order to have a higher probability to sample individual in regions with better evaluations of the objective function, or less constraint violations.

In this paper the Gaussian Mixture Model (GMM) [11, 12] has been chouse as the probabilistic model to generate the offspring population for each iteration. The GMM is defined as a linear combination of n Gaussian Probability Density Functions (PDFs) with non-negative mixing proportions or weights π_i. This is applied to each component density p_i of the mixture.

$$p(\vec{x}) = \sum_{i=1}^{n} \pi_i \cdot p_i = \sum_{i=1}^{n} \pi_i \cdot N(\vec{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \qquad \text{with} \quad \sum_{i=1}^{n} \pi_i = 1, \quad 0 \leq \pi_i \leq 1 \quad (1)$$

Each Gaussian component of the model is characterized by its own mean and covariance matrix. In the EDA, a Gaussian component is defined for each population individual. The component is centered on its individual by setting $\mu = \vec{x}$. Then, diagonal covariance matrix Σ with size $n_{dim} \times n_{dim}$ is parametrically defined for each component of the GMM.

In this work, the mixing weights π_i are related to the objective function evalutations to lead the algorithm toward the optimal solutions. Moreover, the EDA employs a dynamic covariance adaptation approach. The main idea is to promote the exploration in the initial phases of the optimization procedure and the exploitation in the last iterations. By defining σ_j as the diagonal term of the covariance matrix Σ and k_{max} as the maximum number of iterations, it is possible to parametrically establish both a maximum and a minimum value for the covariance.

$$\sigma_{j,max} = \alpha \frac{\left|x_j^{ub} - x_j^{lb}\right|}{m} \; ; \qquad \sigma_{j,min} = \alpha \frac{\left|x_j^{ub} - x_j^{lb}\right|}{m \cdot k_{max}} \quad (2)$$

In this study, to improve the optimization process toward the optima, the mixing weights π_i of GMM are related to the OF of each candidate solution in the design space. Denoting k_{max} as the maximum number of iterations, in order to promote exploration in the early beginning of the iterative optimization process, and promoting exploitation towards the end of the iterative process, the current EDA adopts a dynamic covariance adaptation approach. Denoting with σ_j the variance diagonal term of the covariance matrix Σ referred to the design variable j, the σ_j has been bounded between an admissible range

$$\sigma_{j,max} = \alpha \frac{\left|x_j^{ub} - x_j^{lb}\right|}{m} \; ; \qquad \sigma_{j,min} = \alpha \frac{\left|x_j^{ub} - x_j^{lb}\right|}{m \cdot k_{max}} \tag{3}$$

Within these limits, continuous dynamic decreasing law of variation for σ_j has been formulated:

$$\sigma_j(k) = \sigma_{j,max} - \frac{\sigma_{j,max} - \sigma_{j,min}}{k_{max}^{\beta}} k^{\beta} \tag{4}$$

In the equations $\alpha > 0$ and $\beta > 0$ are user-defined hyperparameter that can be used to heuristically control the exploration and exploitation phases.

The GMM defined in this manner is employed, iteration by iteration, to generate offspring samples and guide the population towards optimal solutions. Unfeasible points are managed using an adaptive penalty-based approach. This approach takes into account: • The degree of constraints violation in candidate solutions; • The number of violated constraints; • The Euclidean distance from the nearest feasible point.

The stopping criteria encompass a predefined maximum number of iterations, a stagnation criterion where the objective function doesn't improve beyond a certain tolerance, and a limit on the computational time.

The EDA algorithm has been implemented in both MATLAB [13] and C#, with the aim of creating a tool for use within the Grasshopper software [14], making it accessible for practitioners.

3 Case study and results

In this paper, the population-based parametri EDA is compared with other metaheruistic algorithms to evaluate the performance of the proposed methodology. In particular, the comparison is made considering the average and the standard deviation of of 20 different optimization runs. The EDA is compared to Genetic Algorithm (GA) and Simulated Annealing (SA). The comparison algorithms have been choose considering that they are already implemented in the Grasshopper software. Thus, the idea is to provide a comparison for the praticitioners that want to perform optimization of the parametrically designed structures within Grasshoper. The case study used for the algorithms comparison is a parametric parabolic arch structure. The Arch is characterized by a free-span $L = 100m$ and a fully restrained static scheme. The objective function to be minimized is the quantity of structural material required to construct the arch. The structure is calculated by using a Finite Element model developed in Karamba3D. The structure is calculated considering the self-weight and a uniform distributed load $q_0 = 250kN/m$. The FEM is employed in the definition of the optimization constraints. In particular, considering a structural steel S355, constraints are defined to ensure that the Von Mises stress remains below the yield strength of the steel, denoted as $\sigma_{VM} < f_y$. Different configurations are defined to prove the effectiveness of the method with an increasing number of design variables. In Figure 1, the different configurations are defined by changing the height in the mid-span, the cross-section shape and allowing a cross-section tapering along the arch axis. The number of the design variables for the studied configurations ranges from 2 to 5. In particular, the optimization design variables are: the height in the mid-span f, the cross-section dimensions r_1, r_2, the cross-section thickness t and the degree of tapering η.

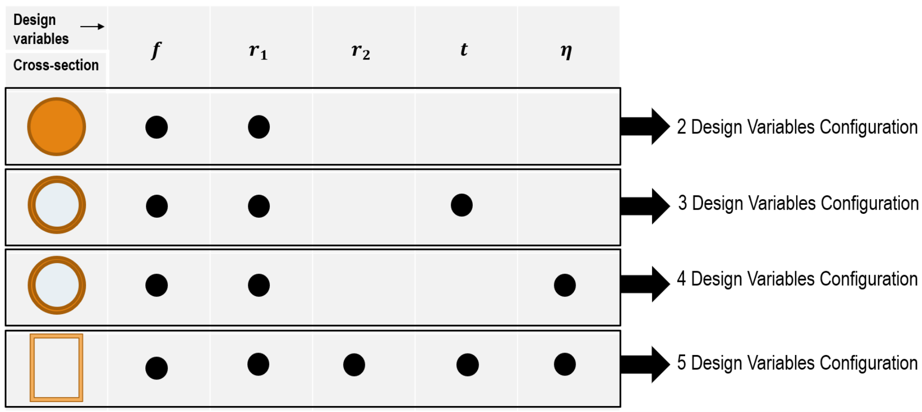


Fig. 1: Different arch configurations with an increasing number of design variables

The optimization procedure is carried out 20 times for each algorithm to assess the robustness of each approach. Figure 2 illustrates the progression of the average objective function value in the 20 runs across the iteration, for all three algorithms. In Figure 3, the evolution of the standard deviation value across the iterations is depicted. Moreover, in Figure 4 the results of the optimization procedure are reported. The results are presented in relation to the design variables, including both the average optimized objective function and its standard deviation. In the presented graphs it is evident that the EDA converges faster to the optimal solutions in most of the cases. Furthermore, it seems to be the most robust algorithm as it is the one with the lowest objective function and standard deviation in three out of four case studies. Additionally, as seen in Figure 4, the EDA demonstrates increasing reliability, especially as the number of design variables grows, in comparison to the other two methodologies.

Finally, in Figure 5 a sensitivity analysis with respect to the two hyperparameters α and β is reported. The graphs refer to the configuration characterized by 5 design variable. In Figure 5a, we observe that for a low number of iterations, the best values for the objective function and standard deviation are achieved with the highest hyperparameter. Considering Figure 5b, it becomes evident that as the number of iterations increases, the optimal configuration is characterized by $\alpha = \beta = 0.5$. By adjusting the hyperparameter values, the user can choose between emphasizing exploration or exploitation. Specifically, when the hyperparameters have high values, the preference is for the exploration phase. The outcome is the attainment of better objective function values in the early stages. In contrast, setting low values of the hyperparameters emphasizes the exploitation phase. The result is that with an increase in the number of iterations, better objective function values can be achieved.

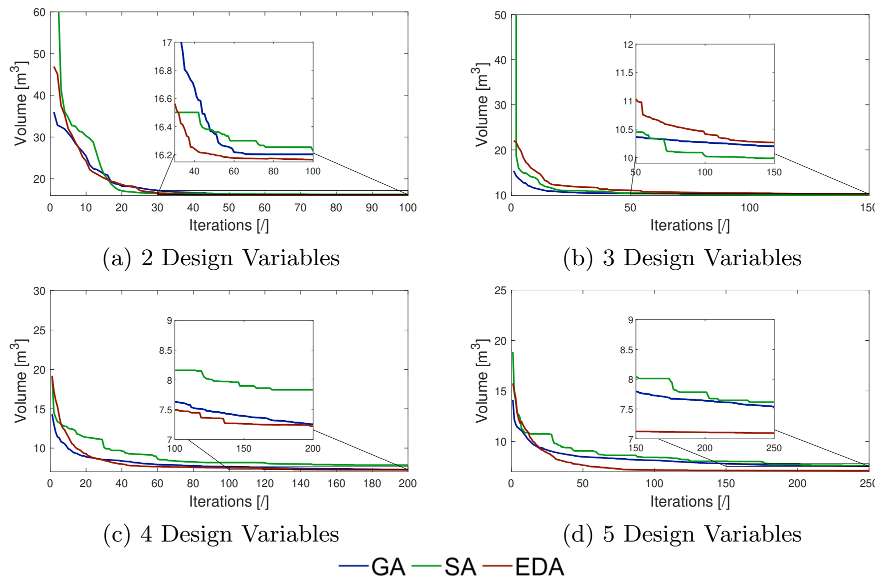


Fig. 2: Comparison of average objective function value evolution through the iterations for the different configurations.

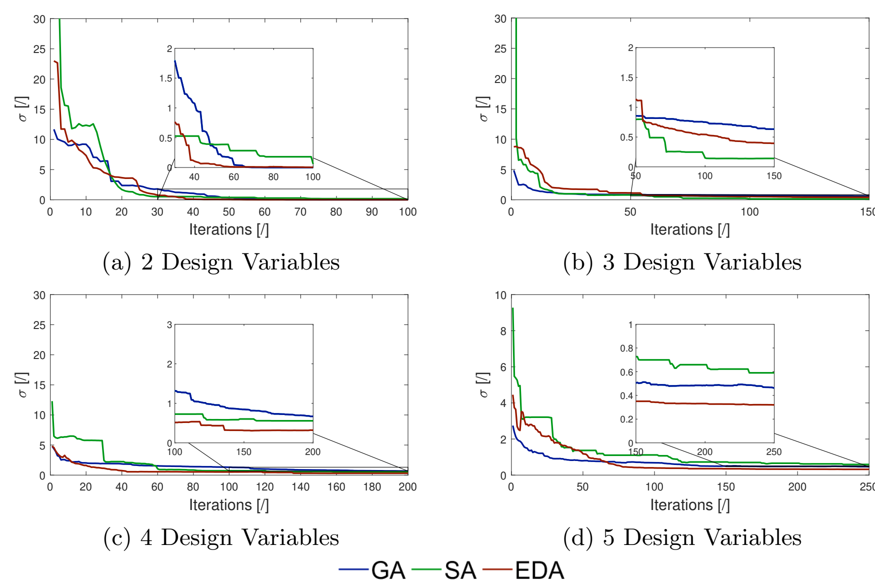
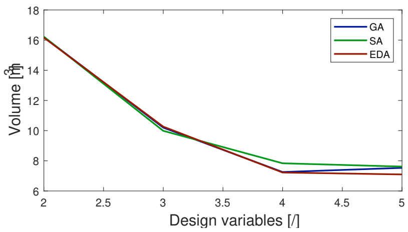
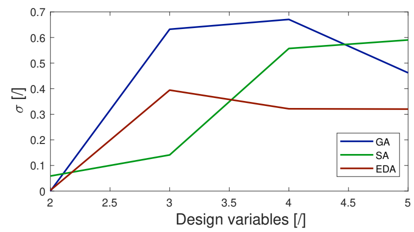


Fig. 3: Comparison of standard deviation value evolution through the iterations for the different configurations.

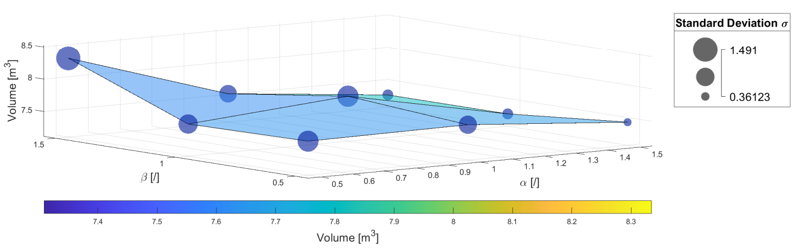


(a) Objective function

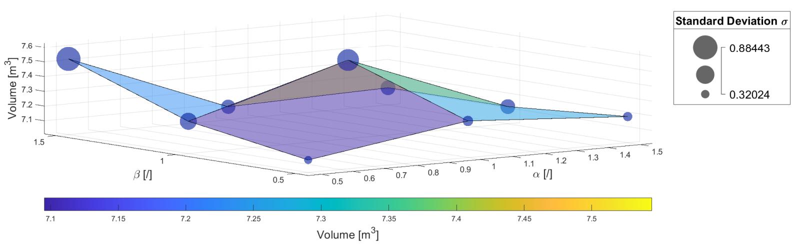


(b) Standard deviation

Fig. 4: Comparison in terms of final objective function and standard deviation for an increasing number of design variables.



(a) 50 iterations



(b) 250 iterationn

Fig. 5: Configuration with 5 design variables: evaluation of objective function and standard deviation for different values of the hyperparameters α and β.

4 Conclusions

This paper introduces a metaheuristic optimization algorithm for parametric structural design optimization, specifically a parametric, population-based Estimation of Distribution Algorithm (EDA). This EDA is implemented in both MATLAB and C# to create a tool compatible with the Grasshopper Software. The paper includes a comparative analysis of the EDA against Genetic Algorithm (GA) and Simulated Annealing (SA), using case studies involving parametric arch structures with an increasing number of design variables. Additionally, a sensitivity analysis of the algorithm concerning hyperparameter values is conducted. The results demonstrate the EDA's superior performance in terms of convergence speed, precision, and reliability compared to GA and SA.

References

1. R. Cucuzza, M. M. Rosso, A. Aloisio, J. Melchiorre, M. L. Giudice, G. C. Marano, Size and shape optimization of a guyed mast structure under wind, ice and seismic loading, Applied Sciences 12 (10) (2022) 4875.
2. G. C. Marano, M. M. Rosso, J. Melchiorre, Optimization as a tool for seismic protection of structures, in: World Conference on Seismic Isolation, Springer, 2022, pp. 100–113.
3. J. Melchiorre, A. Manuello, F. Marmo, S. Adriaenssens, G. Marano, Differential formulation and numerical solution for elastic arches with variable curvature and tapered cross-sections, European Journal of Mechanics-A/Solids 97 (2023) 104757.
4. J. Melchiorre, A. Manuello, L. Sardone, G. C. Marano, et al., Damaging configurations in arch structures with variable curvature and tapered cross-section, WCCM-APCOM 900 (2022) 2022.
5. A. Manuello Bertetto, J. Melchiorre, L. Sardone, G. Marano, et al., Multi-body rope approach for the form-finding of shape optimized grid shell structures, in: Pursuing the Infinite Potential of Computational Mechanics, International Centre for Numerical Methods in Engineering, CIMNE, 2022.
6. J. Melchiorre, A. M. Bertetto, G. C. Marano, Application of a machine learning algorithm for the structural optimization of circular arches with different cross-sections, Journal of Applied Mathematics and Physics 9 (5) (2021) 1159–1170.
7. M. M. Rosso, J. Melchiorre, R. Cucuzza, A. Manuello, G. C. Marano, Estimation of distribution algorithm for constrained optimization in structural design, WCCM-APCOM 2022 900 (2022).
8. M. Pelikan, M. W. Hauschild, F. G. Lobo, Estimation of distribution algorithms, Springer handbook of computational intelligence (2015) 899–928.
9. J. H. Holland, Genetic algorithms, Scientific american 267 (1) (1992) 66–73.
10. S. Kirkpatrick, C. D. Gelatt Jr, M. P. Vecchi, Optimization by simulated annealing, science 220 (4598) (1983) 671–680.
11. G. J. McLachlan, K. E. Basford, Mixture models: Inference and applications to clustering, Vol. 38, M. Dekker New York, 1988.
12. G. McLachlan, D. Peel, Finite mixture models, New York: Wiley, 2000.
13. MATLAB, version (R2022b), The MathWorks Inc., Natick, Massachusetts, 2022. URL https://www.mathworks.com/products/matlab.html
14. R. McNeel, et al., Rhinoceros 3d, version 7.0 (https://www.rhino3d.com/), Robert McNeel & Associates, Seattle, WA (2023).