



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>
Handle ID: <http://hdl.handle.net/10985/21655>



This document is available under CC BY-NC-ND license

To cite this version :

Waad ALMASRI, Dimitri BETTEBGHOR, Faouzi ADJED, Fakhreddine ABABSA, Florence DANGLADE - GMCAD: an original Synthetic Dataset of 2D Designs along their Geometrical and Mechanical Conditions - Procedia Computer Science - Vol. 200, p.337-347 - 2022

Any correspondence concerning this service should be sent to the repository

Administrator : scienceouverte@ensam.eu



3rd International Conference on Industry 4.0 and Smart Manufacturing

GMCAD: an original Synthetic Dataset of 2D Designs along their Geometrical and Mechanical Conditions

Waad ALMASRI^{a,b}, Dimitri BETTEBGHOR^a, Faouzi ADJED^a, Fakhreddine ABABSA^b, Florence DANGLADE^b

^a*Expleo France, 3 avenue des Prés, Montigny-le-Bretonneux 78180, France*

^b*Laboratoire d'Ingénierie des Systèmes Physiques et Numériques (LISPEN), Arts et Métiers, 2 Rue Thomas Dumorey, Chalon-Sur-Saone 71100, France*

Abstract

We build an original synthetic dataset of 2D mechanical designs alongside their mechanical and geometric constraints, GMCAD. Such a dataset allows training Deep Learning (DL) models for Design for Additive Manufacturing (DfAM) to incorporate and control Computer-Aided-Design (CAD) features with mechanical performance. Geometric AM constraints are often complex to describe, depending on applications, processes, materials. They often lack explicit mathematical descriptions, belong exclusively to the CAD world, and hardly can be integrated into mechanical design, hampering AM design freedom. DL models have recently emerged as a potential to reconcile both CAD and Computer Aided-Engineering (CAE) worlds. They derive data-driven geometric rules over mechanical designs, allowing fine-grained control over the geometry during the design phase, contrary to the conventional CAD-to-CAE sequential approach. DL models, however, need high-quality labeled data, and merging CAD features to CAE aspects is challenging as they rely on different formats, rules, and tools. GMCAD dataset solves this issue following these building steps. (i) Building a DL-mechanical conditions predictor from a dataset generated by a density-gradient-based Topology Optimization method (TO); an AM-synergetic design generation tool. (ii) Creating a CAD dataset inspired by the TO-based designs. (iii) Predicting the mechanical conditions of the CADs using the DL predictor of mechanical conditions. Last, we evaluate the mechanical performance of GMCAD's designs and derive statistics over CAD and CAE features. Designs of GMCAD show the significant influence of minor geometric changes, explaining the intricate design task of conforming both with functionality and geometric constraints. Consequently, having GMCAD is advantageous to train DL models to generate designs accounting for all these constraints simultaneously, without the need for time-consuming trial and error techniques. Such models could enhance DfAM and go beyond AM; they can also enhance other challenging fields as CAD automatic reconstruction, reverse engineering, isogeometric design and paves the way to multi-objective controllable design generation.

© 2022 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the 3rd International Conference on Industry 4.0 and Smart Manufacturing

Keywords: Topology Optimization; Additive Manufacturing; Deep Learning.

* Waad ALMASRI. E-mail address: waad.almasri@ensam.eu Tel.: +33-1-3012-2500

Nomenclature

TO Topology Optimization
AM Additive Manufacturing
DL Deep Learning

1. Introduction

Additive Manufacturing (AM) reconstructs designs in a layer-by-layer manner, contrary to conventional subtractive manufacturing, where the material is removed sequentially until the target shape is built. This layer-by-layer reconstruction made AM further attractive in the industrial world. The complexity of a design is no longer an issue, and optimal layouts (in terms of material and functionality) are not compromised anymore to ensure manufacturability. [22]. Additionally, AM allows mechanisms that further reduce the need for material; hence, the cost of the part fabricated, such as infills[29], multi-degree-of-freedom AM systems that reduce the need for support structures, and the laborious post-processing [15].

On the other hand, topology optimization (TO) allows the generation of organic, lightweight shapes optimized for a given functional objective (compliance, stress, thermal distortion, etc.). This synergy between AM and TO encouraged research into orienting TO towards AM.

Theoretically, AM allows the fabrication of any design, but practically, it has its geometric constraints. Furthermore, since a traditional TO does not consider these constraints inherently, researchers tried to find workarounds to ensure that TO's designs comply with manufacturing rules. The authors of [16] opted to modify the optimal layouts to meet the overhang constraints and obtain printable designs without the need for support structures. However, changing the optimal layout suggested by TO might compromise its intended functionality. Thus, others integrated AM-constraints analytically into TO's formulation. Gaynor and Guest[10] incorporated the overhang constraint via a projection-based method. Zhang et al.[33] reformulated SIMP to account for the overhang and hanging features. They generalized and improved their previous work to consider directional-dependent overhang constraint and minimum length control and generated high-resolution 3D structures [34]. Also, the authors of [6, 12, 18, 33] adapted TO approaches to include overhang limitations and deliver self-supporting and print-ready designs. Yoely et al. [31] proposed an optimization approach constraining the areas of holes and curvatures of boundaries using the B-spline representation to obtain a manufacturable design. Xu et al.[30] integrated AM support structure and thin feature constraints into Bidirectional Evolutionary Structural Optimization (BESO) to obtain AM-friendly designs. Zhou et al.[37] integrated minimum length scale to TO via a computationally efficient Finite-Element(FE)-free filtering-threshold scheme. Chandrasekhar and Suresh[8] ensured length scale control using a frequency projection concept via neural networks (NN), making AM-TO more computationally efficient. However, controlling the maximum and minimum length tended to introduce thin cavities making support removal very difficult. Thus, Fernandez et al.[9] proposed to control not only minimum and maximum member sizes but also the minimum gap between structural members to avoid the presence of a large number of thin features and small cavities. Authors in [17, 19, 28] included the build orientation into TO, especially that an adequate build orientation avoids staircase effects, the need for support structures, and hence the need for extra material, build time, and post-processing.

While these methods have revolutionized AM-oriented-TO by generating printable designs, they still suffered from several setbacks. Some geometrical-related constraints between structural members (e.g., angles between structural members) cannot be easily formulated analytically, and some (e.g., hanging features) lack an exact mathematical description [35]. In [9, 34], several parameters need to be chosen carefully to avoid the introduction of non-linearities and ensure convergence of the optimization problem. Moreover, although some work tried to impose geometric constraints via density-filters [37] to alleviate the computational cost of FE-based-TO, or used NNs' back-propagation [8] to compute expensive sensitivity analysis, they still relied on iterative and computationally expensive FE-solvers and FE-analysis, and sometimes ended up with local optima. Finally, whereas it is less intricate to impose few geometric constraints, it is further complicated to handle numerous ones simultaneously, especially inter-member constraints of a single structure, with the analytical AM-oriented-TO formulations.

Consequently, in this work, we propose to solve the problem via Deep Learning (DL) architectures and particularly convolutional neural networks (CNNs), for they have proven efficient and robust in extrapolating spatial correlations and solving geometric-related problems [11, 20, 36]. Their computational cost, in terms of operations and prediction time, is independent of the inputs' complexity, unlike FE-methods, which grow exponentially with input's complexity. Henceforth, instead of trying to integrate analytically geometric constraints into TO's formulation and generate designs using FE-solvers, we propose to explore DL-architectures to impose member-wise and inter-member geometric AM constraints.

Nevertheless, creating a DL-AM-driven-TO, which grasps the geometrical and mechanical variations in designs, needs a rich training dataset with a wide variety of pairs of these designs alongside their constraints. Creating this dataset is challenging, especially since no FE-based TO method in the literature handles several geometric conditions at once. Thus, to achieve our goal, we propose to resolve the problem inversely (detailed in section 3, Fig. 1).

First of all, we generate designs from given mechanical constraints (boundary conditions, loads configuration, and volume fraction) using a modified version of the open-source code of Solide Isotropic Material with Penalization (SIMP) written by Sigmond [25]. Second, a DL model that maps designs to their corresponding mechanical conditions, using the SIMP designs, is built; the model is referred to as DL-Mechanical-Conditions-Predictor. Third, synthetic designs (inspired by the shapes of SIMP designs) with various geometric constraints are generated using *pygmsh*¹. Fourth, the previously learned DL-Mechanical-Conditions-Predictor is used to predict their mechanical conditions. Finally, the target dataset "GMCAD" is consolidated; it consists of pairs of designs and their mechanical and geometric constraints. Consequently, we can train DL-AM-TO, which integrates mechanical and AM-geometric constraints simultaneously at the conceptual level and allows the user to tailor its input constraints easily and generate designs instantly. It is worth mentioning that, in the AM field, a controllable design generation, combining the mechanical and geometrical natures exhaustively, is recently further explored [21]. In fact, with an adapted synthetic dataset as GMCAD and DL techniques, DL-driven generative design approaches offering to manage several parameters and constraints jointly will motivate the implementation of lighter and faster modules in CAD software.

The major contributions of this article can be summarized as follows: (1) The creation of a synthetic dataset of designs alongside their mechanical and geometrical conditions, which (2) allows the inverse problem resolution regarding generating designs complying with mechanical and geometric constraints simultaneously. (3) Finally, the usage of DL techniques to mitigate the difficulty of integrating geometric constraints at the conceptual level of mechanical design and the expensive computational costs of iterative FE methods [3].

The rest of the paper is organized as follows: section 2 provides a theoretical overview of density-based TO. Section 3 details GMCAD's conception workflow. Section 4 evaluates the DL-Mechanical-Conditions-Predictors, presents and evaluates a sample of the resulting target dataset "GMCAD". Section 5 summarizes the work and discusses the future development of the method.

2. Density-based Topology Optimization

Topology optimization (TO) finds the optimal layout within a design space subject to specific boundary conditions, load configurations, and a volume fraction constraint. It allows effective use of the material and generates organic shapes, which makes it synergetic with AM. In the literature, several approaches were proposed to solve the TO problem such as density-based[4] and level-set[1, 2, 27]. Furthermore, Sigmund and Maute[24] reported and detailed most of TO approaches developed in the state of the art (density-based, level set, phase field, and discrete approaches). This work is based on density-based TO for it relies on the continuous approach and considers the design space as a pixelized image. This approach is more compatible with the convolutional architecture of Deep Learning to be trained in order to integrate geometric and mechanical constraints at the conceptual level of design. The topmost common commercial approach is the density gradient-based Solid Isotropic Material with Penalization (SIMP) method [4]. SIMP represents a design as a distribution of discretized square material elements e . The variables are the element-

¹ *Pygmsh* is a python library to draw shapes in FreeCAD, <https://pygmsh.readthedocs.io/en/latest/>

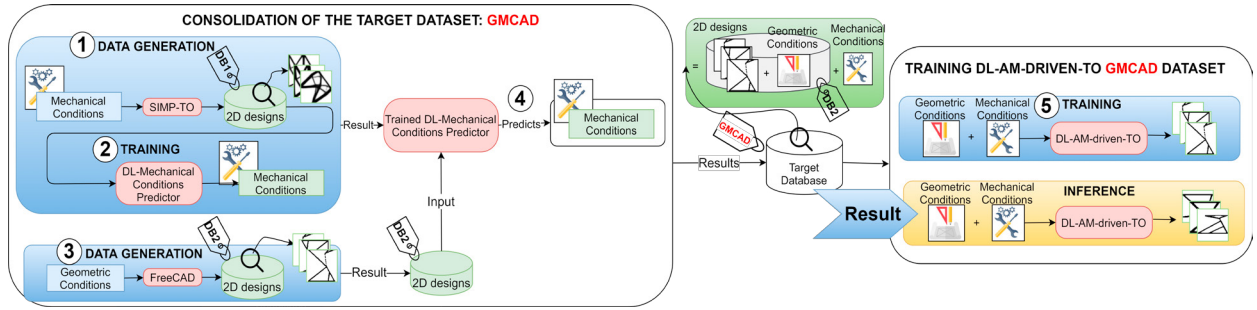


Fig. 1: The workflow is divided into two global steps. The first step consists of consolidating the target training dataset GMCAD. The second step involves training the DL-AM-driven-TO, a DL model that generates designs from input mechanical and geometrical constraints.

relative-densities x_i such that $x_i = 1/0$ represents presence/absence of material at point i of the design domain. A TO problem where the objective is to minimize the compliance $c(x)$ can be written as the following:

$$\min_x = U^T K U = \sum_{e=1}^N x_e^p u_e^T k_0 u_e \quad s.t. \quad K U = F, \quad \frac{V(x)}{V_0} \leq f, \quad 0 < x_0 \leq x \leq 1 \quad (1)$$

where U and u_e are the global and element-wise displacements, F the forces vector, K and k_e are the global and element-wise stiffness matrices and N = number of elements used to discretize the design domain. x_0 is the minimum density material (non-zero to avoid singularity) and p penalization power. V_0 and $V(x)$ are the design domain volume and material volume respectively and f the volume fraction. To solve the problem stated above, Sigmund[25] used the Optimality Criteria (OC) method and added a mesh-independency filter to ensure the existence of solutions to the problem and avoid checker-board patterns[23]. In this study, a modified Python version of the 99-line-of-code of the SIMP method written by Sigmund[25] is used to generate a dataset used to train the Mechanical Conditions Predictors (sections 3.1 & 3.2)².

3. Workflow

As previously mentioned, GMCAD aspires to bridge the mechanics to the geometry at the design level. TO generates layouts given mechanical constraints as inputs. However, it is rather complex for it to handle geometric constraints. Therefore, we choose to resolve the problem inversely. We can always learn to inverse TO's job via DL, i.e., create a DL-Mechanical-Conditions-Predictor that takes a design as input and predicts its mechanical conditions. Then, we create designs (with layouts inspired by SIMP's suggestions) accounting for the desired geometrical constraints, deduce their mechanical constraints using the former DL-Mechanical-Conditions-Predictor. Now that we dispose of a complete dataset with mechanical and geometrical constraints, we can go to the next step and train a DL model that generates designs accounting for the mechanical and geometrical aspects jointly.

The global work is divided into two steps. The first step consists of consolidating the target training dataset GMCAD, which makes the subject of the present article, in an inverse problem resolution way, from the geometry to the mechanics. The second step involves training the DL-AM-driven-TO model and is reported to future work.

The first step is partitioned into four stages: (1) The generation of *DB1*, a dataset of 2D designs from mechanical conditions using SIMP-TO. (2) The training of a DL-Mechanical-Conditions-Predictor to learn to map designs to their corresponding mechanical conditions. (3) The inception of *DB2* a synthetic dataset of CADs (inspired from SIMP-designs' shapes) from geometric conditions. (4) The completion of the synthetic dataset by predicting the mechanical conditions of the CADs. Consequently, the target dataset, GMCAD, is built. It is a synthetic dataset joining designs along with their geometric and mechanical conditions. GMCAD will be used in the future to train the DL-AM-driven-TO model.

² This code is available on the GitHub repository: https://github.com/dbetteb/TOP_OPT.git.

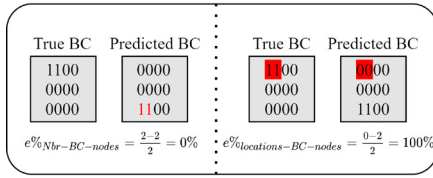


Fig. 2: Example of the evaluation metrics computation of BC-DL-Predictor. This figure compares a predicted to a true BC matrix. The total number of predicted BC nodes is correct, while their locations are erroneous. This error is detected by the metric $e\%locations-BC-nodes$ which counts the number of BC nodes in the correct locations, here none.

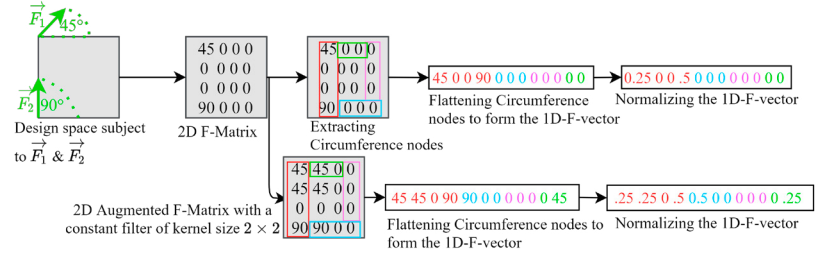


Fig. 3: In this figure, the design space (3×3) is subjected to 2 loads: \vec{F}_1 on the top-left edge with $\theta = 45^\circ$ and \vec{F}_2 on the bottom left edge with $\theta = 90^\circ$. The 2D load matrix (4×4) consists of null-values everywhere except for the loaded nodes set to their orientations. Next, since we only deal with edge-like loads in this work, we reduce the load matrix to the circumference nodes. Finally, the 1D-load vector is normalized (a division by 180). To facilitate the angles prediction, we augment the load matrices by a constant filter of kernel size 5×5 , but for presentation purposes, the filter size shown in the figure is 2×2 .

3.1. Generation of the SIMP dataset DB1

DB1 comes from the mechanical SIMP-TO code and consists of generating 2D designs (as images) from various mechanical conditions (boundary conditions, loads configuration, and volume fraction). For the present work, only edge-like boundary conditions and edge-like punctual loads were considered; i.e., boundary conditions and loads were chosen along the circumference of the design space. Additionally, the maximum number of loads is one. A sample of SIMP designs alongside their mechanical constraints are presented in Fig. 6.

It is important to emphasize that using SIMP-TO, the output layout is independent of the load intensity, and the load orientations are of modulo 180° , i.e., a design subject to a load of 90° is similar to another subject to a load of 270° .

3.2. Mechanical Conditions Prediction

The mechanical conditions that were considered in this work are the volume fraction V , the boundary conditions BC and the loads' configurations F . The volume fraction constraint is the average density values of a design, i.e., the average pixel values in the image-like design. The boundary conditions (BC) and loads (F) are 2D matrices of size $(n_x + 1, n_y + 1)$ for a 2D design of size (n_x, n_y) . BC matrices are 2D matrices with null values everywhere except for the fixed nodes, which are set to 1.0 (Fig. 2 shows an example of a BC matrix of size 4×4). F matrices are 2D matrices with null values everywhere except for the loaded nodes, which are set to θ , where θ is the orientation of the load. Thus, we develop a convolutional DL model that takes 2D designs and reconstructs the BC and F matrices. Instead of predicting BC and F in a single shot, we create a model per constraint; the BC-DL-Predictor and the F-DL-Predictor. The SIMP-based dataset DB1 was split into a train (4538 samples) and a test (1140 samples) set to train the DL models. In this work, n_x and n_y are set to 100, $\theta \in [0^\circ, 180^\circ]$ and $|F| = 1N$.

3.2.1. BC-DL-Predictor

The BC-DL-Predictor's architecture was inspired by the convolutional Res-U-Net architecture [32]. The network is constituted of an encoder, a bridge, and a decoder. The encoder is formed of 4 blocks, each consisting of a down-sampling layer (a convolution of stride 2) and a residual unit³. The decoder comprises 5 blocks, each consisting of an up-sample layer (a transpose convolution of stride 2) and a residual unit followed by a convolution of kernel size 1×1 and a sigmoid activation. The bridge connection has the same architecture as an encoder-block and combines the encoder with the decoder. BC-DL-Predictor takes as input the 2D design and outputs a two-channel matrix corre-

³ A residual unit block is a sequence of two blocks, each consisting of a batch normalization followed by a ReLU activation and a convolution of kernel size $= 3 \times 3$ and stride 1. Its input is summed with its output via an identity mapping connection. An identity mapping connection consists of a convolution of kernel size $= 1 \times 1$, a stride of 1, and padding of 0 followed by a batch normalization layer [32].

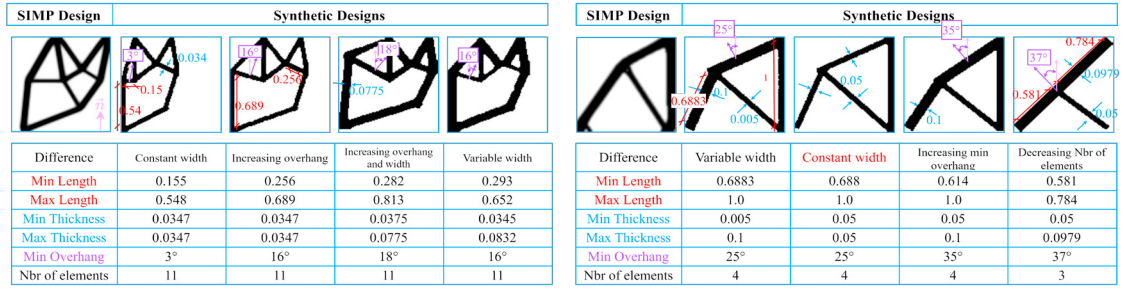


Fig. 4: A sample of the synthetic dataset *DB2*, inspired from SIMP layouts, with various geometric constraints: minimum/maximum element-length, minimum/maximum element-thickness, minimum overhang, and the number of elements. The minimum overhang is computed with respect to the build direction \vec{n} . Here, \vec{n} is along the y-axis. NB: the overhang is the angle between the normal vector of an element (purple arrow) and the build direction \vec{n} (pink arrow).

sponding to the BC along the x- and y-axis, BC_x and BC_y respectively. The model was trained with a learning rate of 0.0002, an Adam optimizer, a batch size of 64, and the mean squared error as a loss function.

3.2.2. F-DL-Predictor

Predicting the loads' matrices is a very intricate task. In the training set, loads matrices are sparse matrices where non-zero values can range from 5 to 180° (loads orientations). Thus, to alleviate the sparsity and wide range of values in the loads' matrices and knowing that we are only dealing with edge loads, we reduce the 101×101 sparse loads matrices to a 1D-normalized-vector consisting of the circumference nodes, i.e., of dimension 1×400 with values range from 0 to 1 (Fig. 3). We note that the minimum load orientation is 5 to avoid confusion with the null values representing the absence of load. Predicting the load location and orientation precisely consisted of two complementary models. The first is the F-DL-Locator model; it predicts with high-precision the locations of the loads. The second, the F-DL-Angle-Estimator model, predicts an augmented version of the load vector. The final load vector is the product of the outputs of the former models (*Load vector* = *Location vector* \times *Augmented Orientation vector*). F-DL-Locator and F-DL-Angle-Estimator share the same architecture, which consists of seven down-sampling layers, each followed by a residual unit [13]. It takes a three-channel input (the 2D design, BC_x , and BC_y) and outputs a 1D vector of dimension 1×400 . Adding the BC as input to the load predictors helps enhance the models' precision for a loaded node is unlikely a fixed one (i.e., a BC node). F-DL-Angle-Estimator was trained with a learning rate of 0.002, an Adam optimizer, a batch size of 64 and the mean squared error as a loss function, and F-DL-Locator with a learning rate of 0.0002, an Adam optimizer, a batch size of 64, and the $L1$ as a loss function.

3.3. Generation of the synthetic geometric dataset *DB2*

The synthetic 2D CAD dataset (*DB2*) is built using pygmsh (a python library, FreeCAD); the CAD shapes are inspired by SIMP-TO's designs. A synthetic design is defined as a connection of beams where we know the four coordinates of every design's beam. This definition of geometry allows us to extract all the geometric information we need from the length, width, overhang (corresponding to a chosen build direction), etc., and consequently obtain a dataset of a wide variety of geometric constraints. Next, the CADs are converted to 2D images. The conversion from CADs to images consists of reading the CAD as a cloud of points, then applying a convolutional filter followed by a bilateral filter to smooth the image without compromising the design's beam thicknesses. *DB2* was generated by varying geometric element-wise (lengths and thicknesses) and inter-element (angles between elements, number of elements) constraints. *DB2* contains all the beams coordinates, lengths, thicknesses, and angles between connected consecutive beams. However, we present here only the geometric-AM-related measures that we will explore (the maximum and minimum element-length, the maximum and minimum element-thickness, the minimum overhang, and the number of elements) with the corresponding designs (Fig. 4). We would like to note that the length and thickness measures are computed with respect to the design space's dimensions (n_x, n_y) (here, $n_x == n_y$, i.e. a length of 1.0 is equivalent to $1.0 \times n_x$).

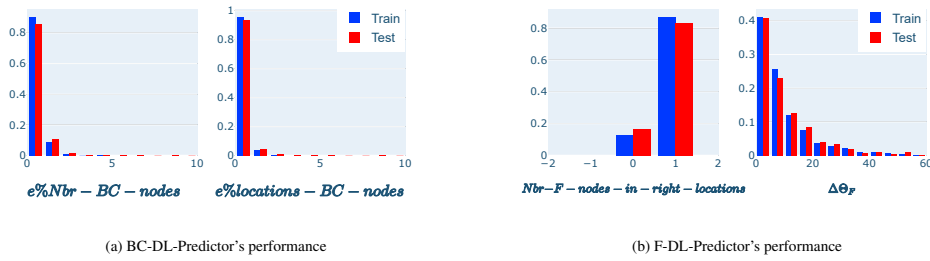


Fig. 5: Mechanical Conditions Predictors Performance over the train and test set.

True BC and F										
Predicted BC and F										
$\Delta Nbr-BC-nodes$	-6	-6	-5	0	0	-5	10	0	0	0
$\Delta Nbr-F-nodes$	0	0	0	0	0	0	0	0	0	0
$\Delta \theta_F$	-91°	4°	-72°	5°	-31°	14°	-2°	-9°	4°	16°

Fig. 6: Comparison between the true and predicted boundary conditions (BC in red) and loads' locations and orientations (F in green). The BC and F locations are predicted with high precision. Most F orientations predictions differ from the true values within $[-20^\circ, +20^\circ]$. NB: The F orientations are in the anti-clockwise direction.

3.4. The Consolidation of the target dataset GMCAD

At this stage, we dispose of *DB2*, a geometric dataset, a BC-DL-Predictor, and an F-DL-Predictor. Consequently, we predict the mechanical constraints of *DB2*'s designs and end up with GMCAD, a dataset with designs along with their mechanical and geometrical constraints. This step consists of first predicting the BC of the designs and then their loads' locations and orientations; since the loads-predictor also needs the BC input. GMCAD's mechanical and geometric-AM-related variables distributions are shown in Fig. 7.

4. Results

4.1. Evaluation of the Mechanical Conditions Predictions

To evaluate the BC-DL-Predictor, we choose two metrics: $e\%Nbr-BC-nodes$ and $e\%locations-BC-nodes$ that calculates the error over the number and location of predicted BC nodes, respectively; the errors vary from 100% (erroneous model) to 0% (accurate model). We would like to emphasize that $e\%Nbr-BC-nodes$ is not sufficient to judge the BC-DL-Predictor's performance, for, in our case, this metric could be deceiving; the model predicts BC nodes in the wrong locations (an example of such case is presented in Fig. 2). Thus, we compute $e\%locations-BC-nodes$ which is the difference of the number of BC nodes between the true and predicted BC matrices in the right locations. This metric ensures that the model well recognizes the BC nodes' locations.

The results of the former metrics are presented in Figure 5a. As we can see, the precision (100%-error) of the BC-DL-Predictor is very high; 83.5% of the test-predictions are detected in the exact locations, and 98% of them within an error margin of 2%.

To evaluate the loads predictions, two metrics were considered: $Nbr_{F-nodes-in-right-locations}$ and $\Delta\theta_F$. $Nbr_{F-nodes-in-right-locations}$ computes the number of loads predicted in the exact location; it measures the precision of the F-DL-locator model. $\Delta\theta_F$ computes the difference between the true and predicted loads orientations in the right loads' locations; the results are illustrated in Fig. 5b. In the test set, the precision of the F-DL-locator is 83.3%, and

only 25% of the angle predictions differs from the true angle by $\pm 20^\circ$.

Although predicting the loads' angles is very intricate, the first results are very encouraging. We would like to point out that improving the precision of the F-DL-Angle-Estimator is still a work in progress. According to our knowledge, we are the first to predict the mechanical constraints from designs via DL models to consolidate a complete dataset of designs with geometric and mechanical conditions altogether.

A sample of designs with their predicted versus true loads and boundary conditions is shown in Fig. 6. As we can see, the BC predictions and loads locations are very accurate. Also, most of the load orientation predictions deviate from the true values within a range of $[-20^\circ, +20^\circ]$.

4.2. “GMCAD”: dataset of designs with their geometrical & predicted-mechanical constraints

GMCAD's variables distributions and a sample of the synthetic designs along their predicted mechanical conditions and performance are shown in figures 7 and 8, respectively. GMCAD contains 36181 distinct combinations of the geometric (Min/Max length/thickness, Min Overhang, Nbr of elements) and mechanical (F orientations and BC nodes) variables. From Fig. 7, we can see that GMCAD comprises a wide variety of designs with 3 to 12 elements, Min bar length $\in [0.07, 1.0]$, Max bar thickness $\in [0.003, 0.4]$, Min overhang $\in [0^\circ, 65^\circ]$, etc. In the future, we could complement GMCAD with designs featuring more than 12 bars, a Max bar thickness ≥ 0.4 , and a Min bar length ≥ 1.0 . The data can be found in [GMCAD's Git repository](#).

From figure 8, we can notice that changing one geometric condition can trigger a difference in the load orientations predictions and its performance (Compliance in Joules J). In figure 8a, the first design's performance twice improved by simply adding an element (2nd design, 12 elements instead of 11). It was further improved when we increased an angle between two bars (the minimum overhang increased) and the thickness of the edge bars; the compliance dropped 16 times (from 412 to 25 J , 5th design). However, increasing the thickness of all bars seemed to deteriorate the functionality of the design (the 3rd design's compliance is 1.5e9 J). The same behavior is detected in Fig 8b. Consequently, the mechanical performance of a design is sensitive to the very minor geometrical change, and achieving the optimal design complying with all constraints takes several geometric modifications. To further understand the influence of geometric constraints on mechanical performance, we generate 16 four-element designs by varying a constraint at a time and compute their compliance (Fig.9). We can observe that the compliance decreases gradually with the non-constant thickness. However, it fluctuates with the minimum overhang and minimum constant thickness. Thus, there is no apparent relation between constant thickness/minimum overhang and the mechanical performance that can be deduced or generalized.

To sum up, several changes in geometric constraints can improve mechanical performance. However, the combinations of constraints are uncountable, and the inter-correlations or dependencies between them are unknown. Moreover, the influence of geometric constraints over mechanical performance cannot be generalized, which encourages the establishment of this dataset and its usage to train a DL model that can jointly capture these correlations and generate designs complying with geometric and mechanical conditions.

5. Conclusion and Future Works

A synthetic dataset of 2D designs alongside their mechanical and geometrical conditions is consolidated in this work. First, we generate mechanical designs using the FE-density-based-TO method (SIMP) and train DL models to predict the mechanical conditions. Then, we build synthetic CADs inspired by the SIMP designs' layouts accounting for various complex geometric constraints, which we complete by predicting the mechanical conditions using the learned DL-Mechanical-Conditions predictors. Finally, we evaluate the synthetic designs' mechanical performance. We showed that even slight geometric changes could deteriorate the design's functionality, and finding the best compromise between mechanical performance and geometry can be very challenging because of the unlimited combinations of geometric modifications and their unknown inter-correlations. Thus, it would be interesting to explore DL in this area for its robustness in capturing spatial correlations and hence create a model that generates designs accounting for mechanical and geometrical constraints concurrently at the conceptual level, DL-AM-TO, which is the next step (Step 5 in the workflow detailed in Fig.1).

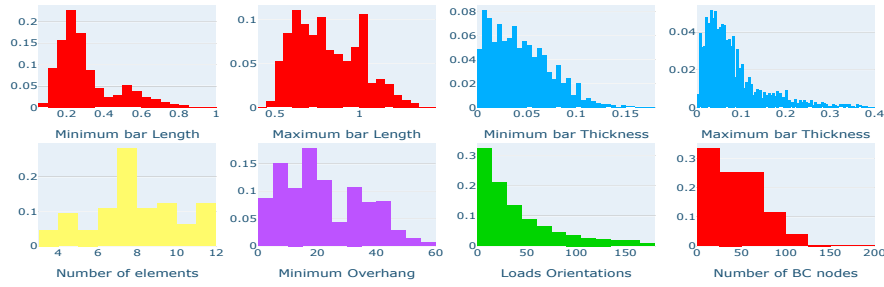


Fig. 7: GMCAD's Descriptive Statistics. This figure shows the distributions of the geometrical (Min/Max Length/Thickness, Number of elements, Min Overhang) and mechanical variables (Number of BC nodes and F orientations) of GMCAD.

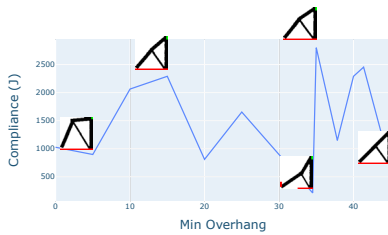
Synthetic Design	$\theta_F=5^\circ$	$\theta_F=5^\circ$	$\theta_F=32^\circ$	$\theta_F=50^\circ$	$\theta_F=74^\circ$
Min Length	0.155	0.155	0.176	0.256	0.293
Max Length	0.548	0.548	0.552	0.689	0.652
Min Thickness	0.0347	0.0347	0.057	0.0347	0.0345
Max Thickness	0.0474	0.0347	0.078	0.0359	0.0832
Min Overhang	3°	3°	3°	16°	16°
Nbr of elements	11	12	12	11	11
Compliance (J)	412	211	1.5 e9	256	25

(a) An 11-element design

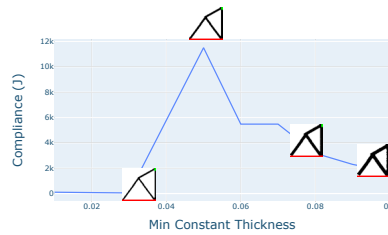
Synthetic Design	$\theta_F=148^\circ$	$\theta_F=85^\circ$	$\theta_F=107^\circ$	$\theta_F=111^\circ$	$\theta_F=108^\circ$
Min Length	0.6883	0.614	0.581	0.688	0.581
Max Length	1.0	1.0	1.0	1	0.784
Min Thickness	0.005	0.05	0.05	0.05	0.05
Max Thickness	0.1	0.1	0.1	0.05	0.0979
Min Overhang	25°	35°	37°	25°	37°
Nbr of elements	4	4	4	4	3
Compliance (J)	55	21	1.1 e9	8.83 e4	1.21 e9

(b) A 4-element design

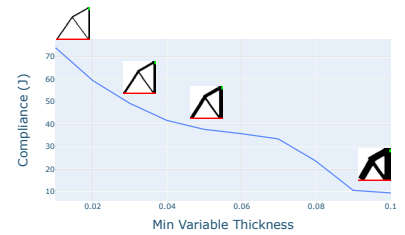
Fig. 8: Synthetic Designs with mechanical and geometrical constraints. In this figure, we show a sample of synthetic designs alongside their predicted mechanical conditions. We also evaluate their mechanical performance (Compliance in Joules).



(a) Compliance versus Overhang



(b) Compliance versus minimum constant thickness



(c) Compliance versus minimum variable thickness

Fig. 9: Mechanical Evaluation of synthetic designs. In this figure, we vary one geometric constraint and evaluate the mechanical performance (Compliance in Joules) of the resultant design. The geometric constraints are the minimum Overhang, the minimum constant bar-thickness (i.e., all elements have the same thickness), and the minimum variable bar-thickness (here, the contour bars have a thickness twice the inner bar).

This article proposes an innovative, unconventional approach to facilitate the integration of complex unformulated geometric AM constraints at the design level without the need for FE iterative expensive computations. Instead of formulating analytically complex geometric AM constraints, it suggests an inverse problem resolution. Designs conforming with these spatial-related constraints are first created, then DL models, particularly CNNs, are trained to learn them. The advantage of this approach is that it can be generalized to incorporate any constraints, even descriptive ones, because we do not need to find a formula for the constraint but simply a sufficient number of examples describing it. Additionally, the usage of this dataset can be extended beyond AM, such as the reverse engineering field, CAD automatic reconstruction, and CAD to CAE. This dataset can also drive DL-multi-objective controllable design generation techniques that can be implemented into CAD software as a lighter and faster generative design module.

Acknowledgements

This work is funded by Expleo France. We would like to thank our intern Corentin MAGYAR for his efforts and help with the diagrams presented in this article.

References

- [1] Allaire, G., Jouve, F., Toader, A. M. (2002) "A level-set method for shape optimization". *Comptes Rendus Mathématique*, 334(12), 1125-1130,
- [2] Allaire, G., Jouve, F., Toader, A. M. (2004), "Structural optimization using sensitivity analysis and a level-set method". *Journal of computational physics*, 194(1), 363-393
- [3] Almasri, W., Bettebghor, D., Ababsa, F., Danglade, F., & Adjed, F. (2021, July). Deep Learning Architecture for Topological Optimized Mechanical Design Generation with Complex Shape Criterion. In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (pp. 222-234). Springer, Cham.
- [4] Bendsøe, M. P. (1989) "Optimal shape design as a material distribution problem". *Structural optimization*, 1(4), 193-202
- [5] Bendsøe, M. P., Sigmund, O. (1999) "Material interpolation schemes in topology optimization". *Archive of applied mechanics*, 69(9-10), 635-654
- [6] Bi, M., Tran, P., Xie, Y. M. (2020) "Topology optimization of 3D continuum structures under geometric self-supporting constraint." *Additive Manufacturing*, 36, 101422
- [7] Booth, J. W., Alperovich, J., Reid, T. N., & Ramani, K. (2016) "The design for additive manufacturing worksheet." In *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers Digital Collection.
- [8] Chandrasekhar, A., Suresh, K. "Length Scale Control in Topology Optimization using Fourier Enhanced Neural Networks. (2020)
- [9] Fernández, E., Yang, K. K., Koppen, S., Alarcón, P., Bauduin, S., & Duysinx, P. (2020) "Imposing minimum and maximum member size, minimum cavity size, and minimum separation distance between solid members in topology optimization." *Computer Methods in Applied Mechanics and Engineering*, 368, 113157.
- [10] Gaynor, A. T., & Guest, J. K. (2016) "Topology optimization considering overhang constraints: Eliminating sacrificial support material in additive manufacturing through design." *Structural and Multidisciplinary Optimization*, 54(5), 1157-1172.
- [11] Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., Lew, M. S. (2016) "Deep learning for visual understanding: A review." *Neurocomputing*, 187, 27-48
- [12] Han, Y. S., Xu, B., Zhao, L., Xie, Y. M. (2019) "Topology optimization of continuum structures under hybrid additive-subtractive manufacturing constraints." *Structural and Multidisciplinary Optimization*, 60(6), 2571-2595
- [13] He, K., Zhang, X., Ren, S., Sun, J. (2016) "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition* pp. 770-778
- [14] Jiang, J., Xiong, Y., Zhang, Z., & Rosen, D. W. (2020) "Machine learning integrated design for additive manufacturing." *Journal of Intelligent Manufacturing*, 1-14.
- [15] Jiang, J., Newman, S. T., & Zhong, R. Y. (2021) "A review of multiple degrees of freedom for additive manufacturing machines." *International Journal of Computer Integrated Manufacturing*, 34(2), 195-211.
- [16] Leary, M., Merli, L., Torti, F., Mazur, M., & Brandt, M. (2014) "Optimal topology for additive manufacture: A method for enabling additive manufacture of support-free optimal structures." *Materials & Design*, 63, 678-690.
- [17] Li, S., Yuan, S., Zhu, J., Wang, C., Li, J., Zhang, W. (2020) "Additive manufacturing-driven design optimization: Building direction and structural topology." *Additive Manufacturing*, 36, 101406
- [18] Mass, Y., Amir, O. (2017) "Topology optimization for additive manufacturing: Accounting for overhang limitations using a virtual skeleton." *Additive Manufacturing*, 18, 58-73
- [19] Matos, M. A., Rocha, A. M. A., Costa, L. A. (2020) "Many-objective optimization of build part orientation in additive manufacturing." *The International Journal of Advanced Manufacturing Technology*, 1-16
- [20] Rawat, W., Wang, Z. (2017) "Deep convolutional neural networks for image classification: A comprehensive review." *Neural computation*, 29(9), 2352-2449
- [21] Rupal, B. S., Mostafa, K. G., Wang, Y., & Qureshi, A. J. (2019). "A reverse cad approach for estimating geometric and mechanical behavior of fdm printed parts." *Procedia Manufacturing*, 34, 535-544.
- [22] Saadlaoui, Y., Milan, J. L., Rossi, J. M., & Chabrand, P. (2017) "Topology optimization and additive manufacturing: Comparison of conception methods using industrial codes." *Journal of Manufacturing Systems* 43, 178-186.
- [23] Sigmund, O., Petersson, J. (1998) "Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima". *Structural optimization*, 16(1), 68-75
- [24] Sigmund, O., Maute, K. (2013), "Topology optimization approaches". *Structural and Multidisciplinary Optimization*, 48(6), 1031-1055
- [25] Sigmund, O. (2001) "A 99 line topology optimization code written in Matlab." *Structural & multidisciplinary optimization*, 21(2), 120-127
- [26] Sigmund, O. (2009) "Manufacturing tolerant topology optimization." *Acta Mechanica Sinica*, 25(2), 227-239.
- [27] Wang, M. Y., Wang, X., Guo, D. (2003), "A level set method for structural topology optimization". *Computer methods in applied mechanics & engineering*, 192(1-2), 227-246
- [28] Wang, C., Qian, X. (2020) "Simultaneous optimization of build orientation and topology for additive manufacturing." *Additive Manufacturing*, 101246

- [29] Wu, J., Aage, N., Westermann, R., & Sigmund, O. (2017) "Infill optimization for additive manufacturing—approaching bone-like porous structures." *IEEE transactions on visualization and computer graphics*, 24(2), 1127-1140.
- [30] Xu, B., Han, Y., Zhao, L., & Xie, Y. M. (2020) "Topological optimization of continuum structures for additive manufacturing considering thin feature and support structure constraints." *Engineering Optimization*, 1-22.
- [31] Yoely, Y. M., Amir, O., Hannel, I., Topology and shape optimization with explicit geometric constraints using a spline-based representation and a fixed grid. *Procedia Manufacturing*, 21, 189-196, (2018)
- [32] Zhang, Z., Liu, Q., Wang, Y.(2018) "Road extraction by deep residual u-net". *IEEE Geoscience and Remote Sensing Letters*, 15(5), 749-753
- [33] Zhang, W., Zhou, L. (2018) "Topology optimization of self-supporting structures with polygon features for additive manufacturing." *Computer Methods in Applied Mechanics and Engineering*, 334, 56-78
- [34] Zhang, K., Cheng, G. , & Xu, L. (2019) "Topology optimization considering overhang constraint in additive manufacturing." *Computers & Structures*, 212, 86-100.
- [35] Zhang, K., & Cheng, G. (2020) "Three-dimensional high resolution topology optimization considering additive manufacturing constraints." *Additive Manufacturing*, 35, 101224.
- [36] Zhao, Z. Q., Zheng, P., Xu, S. T., Wu, X. (2019) "Object detection with deep learning: A review." *IEEE transactions on neural networks and learning systems*, 30(11), 3212-3232
- [37] Zhou, M., Lazarov, B. S., Wang, F., & Sigmund, O. (2015) "Minimum length scale in topology optimization by geometric constraints." *Computer Methods in Applied Mechanics and Engineering*, 293, 266-282.