

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/333677991>

Optimisation par algorithme génétique de structures lattices pour la génération de supports en fabrication additive

Conference Paper · January 2019

CITATIONS

0

READS

20

4 authors, including:



Benjamin Vaissier

Ecole Nationale Supérieure d'Arts et Métiers

4 PUBLICATIONS 2 CITATIONS

[SEE PROFILE](#)



Jean-Philippe Pernot

Ecole Nationale Supérieure d'Arts et Métiers

127 PUBLICATIONS 639 CITATIONS

[SEE PROFILE](#)



Philippe Véron

Ecole Nationale Supérieure d'Arts et Métiers

117 PUBLICATIONS 932 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



MONUMENTUM Project [View project](#)



software development manager [View project](#)

Optimisation par algorithme génétique de structures lattices pour la génération de supports en fabrication additive

B. Vaissier ^{1,2}, J.-P. Pernot ¹, L. Chougrani ² et P. Véron ¹

¹ Arts et Métiers, LISPEM EA 7515, HeSam, Aix-en-Provence, France

² Poly-Shape, 235 rue des Canesteu, Salon-de-Provence, France

Résumé

L'émergence et le développement des technologies de Fabrication Additives (FA) permet la production de formes complexes jusqu'ici inconcevables. Cependant, pour produire ces géométries tortueuses, l'ajout de structures supports est nécessaire. En plus de consommer inutilement de la matière, la fabrication de ces supports nécessite un temps de production et de retrait en finition coûteux. Cet article propose une nouvelle méthodologie d'optimisation de la géométrie et de la topologie des supports internes et externes à une pièce. En partant d'une structure lattice uniformément périodique remplissant à la fois le volume sous les contre-dépouilles intérieures et extérieures, cette approche retire un nombre maximal de poutres afin de minimiser le volume de support. La géométrie la plus adaptée pour la structure lattice initiale est définie au départ en considérant les contraintes du procédé de fabrication. Ensuite, l'élagage du lattice est réalisée grâce à un algorithme génétique pour lequel les paramètres de contrôle ont été optimisés au préalable au travers d'un plan d'expérience. L'approche proposée est validée sur plusieurs cas tests aux géométries variables, contenant à la fois des surfaces intérieures et extérieures en contre-dépouille. Les supports générés sont comparés à ceux obtenus via différentes stratégies proposées dans la littérature et démontrent une consommation de matière moindre.

Mots-clés : supports, structures lattices, algorithme génétique, arbre de Steiner orienté, fabrication additive

1. Introduction

La Fabrication Additive (FA) a fortement évolué vers sa phase d'industrialisation au cours des dernières années, et le domaine prend rapidement de l'ampleur. Cette nouvelle famille de technologies de fabrication rend possible la production de pièces aux géométries complexes, impossible à produire avec les procédés de mise en forme traditionnels. La FA joue donc un rôle clef dans l'émergence de la dernière révolution industrielle, l'Industrie 4.0, qui encourage l'intégration de systèmes de production intelligents et de technologies de l'information perfectionnées [DGYD17]. Contrairement aux procédés soustractifs, la FA consiste à agglomérer de la matière pour créer des objets à partir de modèles 3D, généralement couche après couche [noa12]. Grâce à cette approche, des géométries telles que des structures lattices ou poreuses, des structures organiques générées par optimisation topologique [PAHA18], des pièces présentant des ca-

naux d'écoulement tortueux [BB16] deviennent possible à produire et leur fabrication est facilitée.

Malgré l'intérêt grandissant et la facilité apparente d'implémentation, la production de pièces en FA nécessite quelques précautions. En effet, avec la plupart des technologies de FA, l'ajout de structures supports est requis afin d'assurer la bonne production d'une pièce. Les structures supports peuvent remplir trois fonctions principales [HHY^{*}13] : (i) soutenir les contre-dépouilles, les ponts et les îlots ; (ii) rigidifier la pièce pour éviter les déformations ; (iii) dissiper la chaleur des zones d'accumulation thermique. Une contre-dépouille correspond à une surface formant par rapport au plan horizontal un angle inférieur à une valeur seuil, appelée angle de contre-dépouille, généralement considérée égale à 45° [Tho09]. Un pont est une large surface en contre-dépouille, généralement horizontale, soutenue à ses deux extrémités. Un îlot correspond à un volume de matière qui sera, à une certaine hauteur de fabrication, complètement déconnecté du reste de la pièce et du plateau de fabrication.

Ce papier se concentre principalement sur la fonction de

soutien, c'est-à-dire comment les zones en contre-dépouille peuvent être soutenues. Les contraintes de rigidification et de dissipation ne sont pas directement abordées, mais sont discutées dans la conclusion. Les structures supports peuvent être classifiées en deux grandes catégories :

- Les supports *détachables* (aussi appelés supports externes) sont généralement localisés autour de la pièce, dans des zones accessibles afin d'être retirés lors de la phase de post-production par les opérateurs de finition. C'est le type de supports le plus utilisé.
- Les supports *permanents* (aussi appelés supports internes) font partie intégrante de la pièce finale, et permettent de soutenir les cavités internes et les zones inaccessibles. Ce type de supports est évité au maximum car cela génère un poids supplémentaire dans la pièce finale, et peut empêcher certaines fonctionnalités de la pièce (comme l'écoulement d'un fluide dans un canal interne par exemple).

Pour minimiser le besoin de supports, certaines approches proposent la génération de géométries auto-supportantes [Lan17]. Cependant, dans la plupart des cas pratiques, il n'est pas possible d'éliminer complètement toutes les surfaces en contre-dépouille. L'optimisation de structures supports représente donc un intérêt financier fort pour l'industrie. Pour les supports détachables, trois caractéristiques peuvent être optimisées : le volume, le temps de production et la facilité de retrait. Le volume d'un support impacte directement la quantité de matière consommée pendant la production. Cela affecte aussi le temps de production. Cependant, le temps de fabrication d'une pièce dépend également de la géométrie des supports, car la vitesse de lasage n'est pas la même pour toutes les surfaces du support. Généralement, la vitesse de lasage du contour d'une géométrie est inférieure à celle du remplissage de cette même géométrie, afin d'obtenir un meilleur état de surface extérieur. Enfin, la facilité de retrait des supports diminue le temps de finition, et donc le coût global de production de la pièce. Pour les supports permanents, par définition, seul le volume et le temps de production peuvent être minimisés.

Cet article propose une nouvelle méthodologie pour l'optimisation de supports, qui repose sur l'optimisation discrète d'une structure lattice. Partant d'un lattice uniforme remplissant les espaces vides sous les surfaces intérieures et extérieures à soutenir, la méthodologie développée retire un maximum d'éléments de la structure lattice initiale afin d'optimiser le volume de la structure support (voir figure 1). Les paramètres de la géométrie du lattice initial sont tout d'abord calibrés en considérant les limitations du procédé de fabrication. La topologie de ce lattice est ensuite optimisée en élagueant le lattice grâce à un algorithme génétique. Des étapes de post-traitements sont ensuite réalisées pour préparer le modèle avant impression.

L'intérêt de ce travail est multiple : (i) il permet de minimiser le volume du support généré, afin de réduire le coût

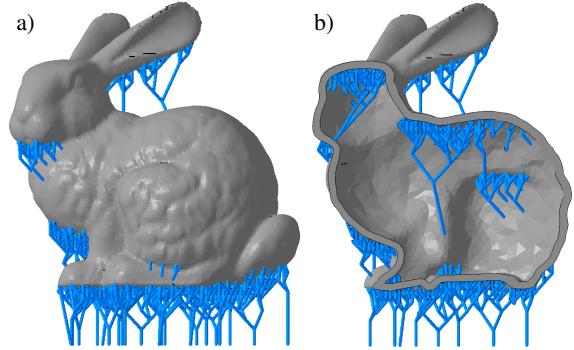


Figure 1: Supports lattices générés à partir d'un modèle triangulé du Lapin de Stanford (Stanford Bunny) en utilisant la méthodologie proposée : vue complète (a) et vue en coupe (b)

global de production de la pièce ; (ii) l'algorithme génère des structures périodiques auto-supportantes en forme d'arbres, sans direction privilégiée, les rendant optimales pour réaliser des optimisations mécaniques et pour leur retrait facile lors des étapes de finition ; (iii) la méthodologie proposée implémente une nouvelle approche utilisant un algorithme génétique pour résoudre le problème d'arbre de Steiner orienté (ou DSTP en anglais) associé à l'optimisation de supports lattices.

Cet article est organisé comme suit. Après une présentation d'ensemble des développements actuels concernant l'optimisation de supports (section 2), la section 3 décrit les différentes étapes de la méthodologie proposée. Le problème de détermination d'un support lattice allégé est ensuite introduit avec l'algorithme génétique (AG) utilisé pour sa résolution (section 4). La méthodologie proposée est ensuite discutée et validée sur plusieurs cas tests, et les résultats sont comparés à ceux obtenus par plusieurs stratégies de supportage identifiées dans la littérature (section 5). Enfin, la section 6 finalise cet article en récapitulant les conclusions et perspectives de ces travaux.

2. État de l'art

Les supports sont essentiels pour la bonne production de pièces en FA. Ils empêchent la matière solidifiée de s'effondrer et réduisent les déformations inhérentes au procédé. Cependant, ils représentent une part importante des coûts de production (c'est-à-dire volume de matière, temps de production et temps de retrait en finition) et leur optimisation est donc cruciale.

Suivant la technologie de FA adoptée, les structures supports ne jouent pas le même rôle, et ne possèdent donc pas les mêmes géométries. Diverses géométries de supports peuvent être identifiées dans la littérature. Elles peuvent être

classifiées en quatres grandes catégories : les structures extrudées, doublement périodiques, triplement périodiques et apériodiques.

Les *structures extrudées* consistent en une forme 2D dans le plan XY répétée à chaque couche jusqu'à la géométrie de la pièce. Ce sont les géométries les plus communes car elles sont faciles à générer et à manipuler. Par exemple, en LBM, Calignano réalise un plan d'expérience pour optimiser les perforations de supports vis-à-vis des déformations de la pièce [Cal14]. Järvinen et al. optimisent la rugosité de la surface de la pièce et la facilité de retrait de tubes et toiles extrudées [JML*14]. Jhabvala et al. génèrent des supports à partir de microstructures poreuses, grâce à un laser pulsé, afin de faciliter leur retrait [JBAG12]. Krol et al. développent une optimisation discrète fondée sur la Méthode des Éléments Finis en subdivisant des murs extrudés qui s'intersectent [KZS12]. Pour la technologie FDM, Jin et al. proposent un algorithme basé sur le slicing afin de générer des supports plastiques extrudés [JHF15] et Huang et al. étendent cette méthode avec un algorithme reposant sur des pixels [HYML09]. Pour la technologie SLA, Quian et al. ont développé un algorithme projetant les surfaces en contre-dépouille sur le plateau d'impression pour générer des supports en forme de bloc [QLYG12]. Enfin, pour la technologie EBM, Cheng et al. et Cooper et al. utilisent des blocs placés sans contact sous les zones en contre-dépouille pour dissiper l'énergie thermique induite par le procédé [CC15] [CSCC15].

Les *structures doublement périodiques* représentent des formes 3D complexes, répétées selon un motif 2D dans le plan XY. Par exemple en LBM, Gan et Wong optimisent des géométries en forme d'arbres en faisant varier la fréquence de répétition dans le plan XY. [GW16]. En FDM, Boyard répète une structure en forme d'arbre sous les contre-dépouilles afin de supporter des pièces plastiques [Boy15].

Les *structures triplement périodiques* représentent des formes 3D complexes, répétées dans les directions X, Y et Z. Hussein et al. utilisent des surfaces minimales pour supporter une poutre cantilever [HHY*13] [Hus13], alors que Cloots et al. empilent des pièces dans la chambre de fabrication en utilisant des supports lattices [CSW13]. En FDM, Li et al. font varier le diamètre de poutres d'un lattice afin de rigidifier leurs supports [LDJ*15], alors que Lee et al. proposent une approche d'évidemment par voxels pour créer des supports internes [LL16].

La catégorie des supports *apériodiques* réunit toutes les géométries de supports qui ne présentent aucun motif de répétition. Par exemple en FDM, Mezzadri et al. génèrent des supports organiques au travers d'une optimisation topologique. De leur côté, Schmidt et al. ont développé un algorithme pour générer des supports en forme d'arbre [SU14]. Cet algorithme a par la suite été implémenté dans le logiciel gratuit Meshmixer. Vanek et al. ont également généré des géométries de supports en forme d'arbres

en calculant l'intersection de cônes placés sous les surfaces en contre-dépouille [VGB14] alors que Vaidya et al. répètent une cellule unitaire octaédrique pour atteindre le même objectif [VA16]. Dumas et al. exploitent les possibilités de la technologie FDM pour créer des ponts afin de réduire le volume de supports [DHL14]. Pour supporter les cavités internes, Mao et al. proposent une approche hybride combinant des structures inspirées de fibres musculaires et des structures triplement périodiques.

La plupart des travaux précédemment mentionnés proposent des algorithmes pour supporter les surfaces en contre-dépouille mais peu d'entre eux prennent en compte l'importance de minimiser le volume des structures générées. En plus de retirer des poutres superflues et donc de minimiser la quantité de matière solidifiée, notre méthodologie manipule une structure lattice pré-optimisée dont le volume est minimisé tout en assurant sa fabricabilité. Par ailleurs, de nombreuses approches explorent l'utilisation de structures extrudées, doublement ou triplement périodiques, mais peu d'entre elles s'intéressent à l'utilisation de supports apériodiques. Cependant, les contraintes de contre-dépouilles et de déformations thermo-mécaniques associées à un problème de supportage ne suivent, la plupart du temps, aucun motif périodique. Ainsi, les structures supports ne devraient à priori pas présenter de répétition de forme. En retirant les poutres superflues d'un lattice, La méthodologie présentée génère des structures apériodiques en forme d'arbre, ne présentant aucun biais de répétition.

3. Méthodologie d'optimisation

Cette section introduit la nouvelle méthodologie d'optimisation développée pour soutenir les surfaces en contre-dépouille d'une pièce. On rappelle qu'une surface est en contre-dépouille si elle forme un angle inférieur à un certain seuil par rapport au plan horizontal. Ce seuil est appelé angle de contre-dépouille, et sa valeur est généralement considérée égale à 45° .

Classiquement, une surface en contre-dépouille est soutenue si chacun de ses points \mathbf{p}_0 est situé à une distance inférieure à une distance de contre-dépouille maximale o_p (p signifiant ici "process") d'au moins un autre point \mathbf{p}_1 avec un support directement en-dessous (\mathbf{p}_1 est donc nécessairement à la frontière de la surface en contre-dépouille). La distance de contre-dépouille maximale dépend de la technologie de FA adoptée, du matériau et des paramètres d'impression. Par exemple, pour la technologie FDM, la distance de contre-dépouille maximale peut atteindre quelques dizaines de centimètres [DHL14] (suivant le matériau) alors qu'en LBM, on considère généralement que $o_p \simeq 0.5\text{mm}$. La figure 2 illustre cela, en représentant les surfaces soutenues en vert, pour lesquelles les points sont suffisamment proches des structures supports existantes. A l'opposé, les zones en rouge représentent les surfaces en contre-dépouille éloignées

de tout support. La condition de soutien peut être exprimée comme suit :

$$\forall \mathcal{S} \in \mathbb{O}(\mathcal{P}), \forall \mathbf{p}_0 \in \mathcal{S}, \exists \mathbf{p}_1 \in \Psi(\mathcal{S}) : \|\mathbf{p}_0 - \mathbf{p}_1\| \leq o_p \quad (1)$$

où \mathcal{S} correspond à la surface en contre-dépouille, $\mathbb{O}(\mathcal{P})$ représente l'ensemble des surfaces en contre-dépouille de la pièce \mathcal{P} (\mathbb{O} pour *overhang* en anglais), et $\Psi(\mathcal{S})$ est l'ensemble des points soutenus de \mathcal{S} possédant un support directement en-dessous.

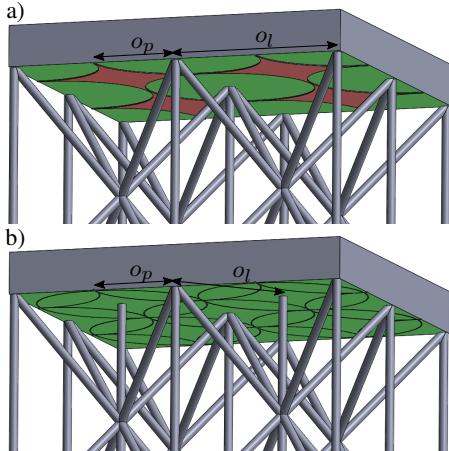


Figure 2: La condition de soutien : une surface partiellement (a) et totalement (b) soutenue. Les zones non-soutenues (en contre-dépouille) sont en rouge.

A partir de cette définition, il devient clair que l'utilisation de structures lattices est un bon moyen d'assurer la condition de soutien tout en minimisant le volume de supports à utiliser en-dessous des surfaces en contre-dépouille. Le principe de la méthodologie développée est de générer un lattice initial sous les surfaces en contre-dépouille d'une pièce, et de retirer de ce lattice le maximum de poutres, sans enfreindre la condition de soutien. Plus précisément, la méthodologie proposée est composée de plusieurs étapes illustrées figure 3 :

1. *Génération du lattice initial* : en partant d'un maillage fermé de triangles composé d'un ou plusieurs solides distincts, un lattice uniforme est généré par la répétition régulière d'une maille élémentaire parallélépipédique fabricable, dans les 3 directions de l'espace 3D, avec 3 pas de répétition constants. Dans le cadre d'application de la méthodologie proposée, une géométrie spécifique pour la maille élémentaire est utilisée, mais toute autre géométrie composée de poutres auto-supportantes aurait pu être choisie. La maille élémentaire adoptée combine une maille Cubique Centrée (ou *Body-Centred Cubic* en anglais) et 5 poutres

verticales, localisées à chaque arête verticale, ainsi qu'en place de l'axe verticale du cube (voir figure 4). Elle est définie par 3 paramètres : a correspond à la taille de la base, h à la hauteur de la maille, et d au diamètre des barreaux. En conséquence, l'angle maximum des poutres α_{max} et la distance de contre-dépouille maximale du lattice o_l (l faisant référence à "lattice") peuvent être facilement calculés. Ce lattice doit satisfaire la condition de soutien (1) car l'algorithme d'optimisation identifie le meilleur sous-lattice du lattice initial. Ainsi, la condition de soutien pour cette maille élémentaire spécifique est $o_l \leq 2o_p$ ce qui donne :

$$a \leq (d + 2o_p)\sqrt{2} \quad (2)$$

Le lattice est ensuite ajusté sur la surface de la pièce : chaque poutre traversante est élaguée afin que son extrémité repose exactement sur l'enveloppe de la pièce. Les paramètres a , h et d de génération du lattice ne sont pas considérés comme des variables de l'optimisation discrète de support lattice.

2. *Pré-traitement* : les variables et paramètres du problème d'optimisation sont déterminés. Les nœuds de connexion du lattice connectés à une seule poutre, aussi appelés "nœuds isolés" de valence 1, sont identifiés. Ces nœuds apparaissent à cause de l'élagage du lattice initial au contour de la pièce. Parmi les nœuds isolés, ceux connectés à une surface en contre-dépouille sont appelés nœuds sources et sont ceux qui doivent être soutenus. Les autres nœuds isolés, qui ne nécessitent pas d'être soutenus, sont appelés puits.
3. *Pré-optimisation* : les variables du problème d'optimisation pour lesquelles la valeur dans la solution optimale peut être déduite sont recensées et retirées de l'ensemble des variables d'optimisation. Cela permet de réduire le temps de calcul. Par exemple, si un nœud source ne possède qu'une seule poutre sortante, cette poutre est nécessaire à la préservation de la contrainte de soutien, et fait donc obligatoirement partie de la solution optimale. Cette poutre est alors directement ajoutée à l'ensemble des poutres formant la solution, la source est retirée de l'ensemble des variables, et l'extrémité inférieure de la poutre est ajoutée en tant que nouvelle source à l'ensemble des variables du problème. De la même façon, si la plus petite poutre sortante d'un nœud est directement reliée à un puit (c-à-d son extrémité inférieure est un puit), alors cette poutre est ajoutée à l'ensemble des poutres solution et la source est retirée de l'ensemble des variables.
4. *Optimisation* : un algorithme génétique (ou *Genetic Algorithm (GA)* en anglais) est utilisé pour déterminer quel sous-lattice du lattice initial possède la plus petite longueur cumulée de poutres (somme des longueurs de chaque poutre) tout en respectant la contrainte de

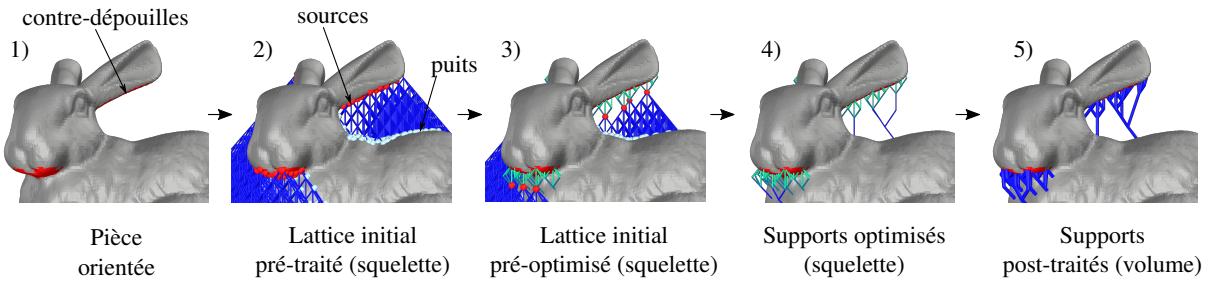


Figure 3: La méthodologie d'optimisation proposée (application à la génération de supports externes pour le Lapin de Stanford)

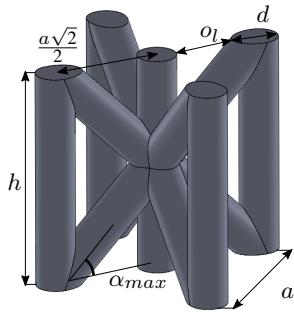


Figure 4: La maille élémentaire utilisée par la méthodologie d'optimisation développée

soutien. Lors de cette étape, le lattage initial est élagué jusqu'à ce que la condition d'arrêt soit atteinte. Cet algorithme est introduit dans le paragraphe 4.

5. Post-traitement : une fois le sous-lattice solution déterminé (dans l'ensemble fini des solutions définies par le lattage initial), une phase de post-traitement est appliquée pour réduire encore un peu plus sa longueur (dans un ensemble de solutions différent, continue dans l'espace). Les chemins de poutres reliant 2 points de connexion (c-à-d les sources, les trous et les points de valence 2 ou plus) sont redressés en remplaçant chaque chemin de poutres par une unique poutre rectiligne entre les 2 points de connexion associés. Grâce à la topologie du lattage initial, la nouvelle poutre ajoutée est assurée d'être auto-supportante, et ne générera donc pas de nouvelle surface en contre-dépouille. Ensuite, la structure lattage résultante est triangulée, en utilisant par exemple l'approche de Chougrani et al. [CPVA17] pour minimiser le nombre de triangles générés.

À la fin de ce processus d'optimisation, les étapes classiques de préparation restantes sont tout de même réalisées avant l'impression : la géométrie générée est tranchée en un

fichier de trajectoires, ce fichier est envoyé à la machine et la paramétrage d'impression est sélectionnée.

4. Optimisation discrète de supports lattes (LS²DO)

Le problème d'optimisation de supports lattes introduit dans la section précédente est un problème discret, car le nombre de solutions potentielles est fini. Pour le résoudre, un graphe acyclique orienté (ou Directed Acyclic Graph (DAG) en anglais) est associé au lattage initial et un Algorithme Génétique (AG) est utilisé pour déterminer le sous-graphe respectant la contrainte de soutien qui possède la plus petite longueur cumulée de poutres.

4.1. Formalisation du problème

Inspiré par la théorie des graphes, un lattage $L = (N, B)$ est un ensemble B de poutres connectées à un ensemble N de noeuds. Le nombre de poutres est noté $b = \text{card}(B) = |B|$ et le nombre de noeuds $n = \text{card}(N) = |N|$. Pour un lattage L , l'étape de pré-traitement de la méthodologie proposée identifie l'ensemble des noeuds sources N_S et l'ensemble des noeuds trous N_W . Un lattage $L' = (N', B')$ est défini comme un sous-lattice de L si $B' \subset B$. Cela implique également $N' \subset N$. Notons $\text{sub}(L)$ l'ensemble des sous-lattes de L .

Le graphe lattage associé à un lattage $L = (N, B)$ est le graphe $G = (V, E)$ dans lequel chaque sommet dans V correspond à un noeud dans N , et chaque arête dans E correspond à une poutre dans B . Ainsi, $\text{card}(V) = |V| = n$ et $\text{card}(E) = |E| = b$. Un élément de V est noté v_i avec $i \in \{1, \dots, n\}$. Similairement, un sommet source v_{Si} est associé à un noeud source du lattage, et un sommet trou v_{Wi} est associé à un noeud trou. De même, V_S représente l'ensemble des sommets sources avec $|V_S| = n_s$, et V_W représente l'ensemble des sommets trous avec $|V_W| = n_W$. Pour un sommet v_i de V , $\text{in}(v_i)$ et $\text{out}(v_i)$ représentent respectivement les ensembles des noeuds entrants et sortants du sommet v_i . Notons out_{max} le nombre maximal de poutres sortantes pour un noeud dans

V , soit $out_{max} = \max_{i=1..n}\{|out(v_i)|\}$. Notons aussi V_{out} , de cardinal n_{out} , l'ensemble des sommets dans V avec au moins une arête sortante : $V_{out} = \{v_i \in V : out(v_i) \neq \emptyset\}$. Du fait que chaque arête est nécessairement une arête sortante pour une de ses extrémités, le nombre de poutres de G peut s'écrire comme suit :

$$|E| = b = \sum_{v \in V_{out}} |out(v)| \quad (3)$$

Dans l'approche développée, le graphe lattice est orienté comme suit. Chaque arête du graphe est orientée de son sommet correspondant au nœud du lattice le plus haut, (suivant Z, perpendiculairement au plateau de fabrication) vers le sommet correspondant au nœud le plus bas. De plus, le graphe lattice est pondéré avec une fonction $w_e : E \rightarrow \mathbb{R}$ qui associe à chaque arête du graphe lattice un coût égal à la longueur de la poutre correspondante. Pour une arête e_j avec $j \in \{1, \dots, |E|\}$, le coût $w_e(e_j)$ peut donc faire référence à la longueur de la poutre correspondante dans le lattice. Ainsi, le poids (ou longueur) $w(G)$ d'un graphe lattice $G = (V, E)$ est donné par l'équation :

$$w(G) = \sum_{j=1}^{|E|} w_e(e_j) \quad (4)$$

C'est la fonction objectif à minimiser lors de la résolution du problème LS²DO. En effet, la méthodologie proposée commence par générer une structure lattice L sous les zones en contre-dépouille (frontière rouge sur le Lapin de Stanford de la figure 5) et optimise cette structure en trouvant un sous-lattice dans $sub(L)$ avec la longueur cumulée de poutres minimale, qui soutien toutes les sources de N_S identifiées. En terme de graphe, le problème est de trouver un sous-graphe du graphe lattice initial $G = (V, E)$ de coût minimum $w(G)$, et qui connecte chaque sommet source dans V_S à au moins un sommet puit dans V_W . La figure 5 montre le graphe acyclique orienté (DAG) associé au problème LS²DO (en gris). Les sommets en rouges sont les sources, les sommets en bleus sont les puits.

Par la suite, si l'on connecte tous les sommets puits à un nouveau sommet racine r , en ajoutant simplement un ensemble d'arêtes E' pondérées avec une valeur de 0 (bleues dans la figure 5), le problème reste inchangé, mais peut être formulé comme suit : "Trouver l'arbre $T = (V_T, E_T)$ dans $G_{ini} = (V \cup \{r\}, E \cup E', w_e)$ de coût minimal $w(T)$, enraciné en r tel que $V_S \subseteq V_T \subseteq V$ et $E_T \subseteq E'$ ". Ainsi, sur la figure 5, les poids des arêtes sont égaux aux longueurs des poutres dans la structure lattice, sauf les arêtes bleues qui possèdent toutes un poids nul et qui ne sont associées à aucune poutre du lattice.

La formulation précédente correspond à la définition du problème de l'arbre de Steiner orienté (ou Directed Steiner Tree Problem (DSTP) en anglais) connu pour être NP-difficile [HK13]. Halperin et al. ont prouvé que, pour tout

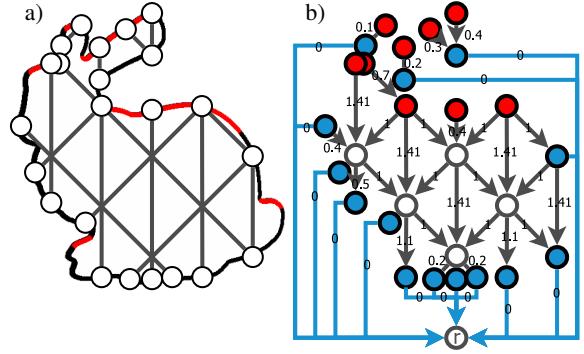


Figure 5: Exemple du problème LS²DO sur le lapin de Stanford en 2D (a) et le graphe acyclique orienté (DAG) pondéré correspondant (b).

$\epsilon > 0$, il n'existe pas d'algorithme d'approximation polynomial dans un ratio de $\log^{2-\epsilon} n$, à moins que $P = NP$ [HK03]. Ainsi, de nombreux sujets de recherche se concentrent sur la découverte d'une solution d'approximation du DSTP [CCC*00] [Zel97].

4.2. Résolution par algorithme génétique

La résolution du problème LS²DO est réalisée en utilisant un algorithme génétique dont les paramètres de contrôle ont été calibrés au travers d'un plan d'expérience. L'algorithme génétique adopté dans la méthodologie proposée a été implémenté grâce à la librairie GeneticSharp [Gia18].

Un algorithme génétique manipule des populations de chromosomes. Chaque chromosome est un ensemble de variables, appelées gènes, encodant une solution potentielle au problème, c-à-d un sous-graphe dans le cas présent. Les valeurs que peut prendre un gène particulier sont appelées allèles. Pour résoudre le problème LS²DO, un encodage pas aiguillage a été utilisé. Ainsi, une variable x_i est associée à chaque sommet v_i dans l'ensemble V_{out} du graphe initial, sélectionnant la seule arête sortante de $out(v_i)$ qui sera activée si au moins une arête entrante de $in(v_i)$ est activée. Le *chromosome d'aiguillage* contient donc $|V_{out}| = n_{out}$ gènes, et le gène x_i prend des valeurs dans $\{1, \dots, |out(v_i)|\}$. Le nombre de chromosomes possibles définis par l'*encodage par aiguillage*, noté C_{switch} , est :

$$|C_{switch}| = \prod_{v \in V_{out}} |out(v)| \quad (5)$$

Dans l'exemple 2D de la figure 6 (à gauche), le gène x_i ne peut prendre que trois valeurs, c-à-d. $out_{max} = 3$. Si $x_i = 1$, alors l'arête orientée partant de v_i et dirigée vers la gauche est activée, sinon dans le cas où $x_i = 2$ alors l'arête verticale orientée vers le bas est activée, tandis que si $x_i = 3$

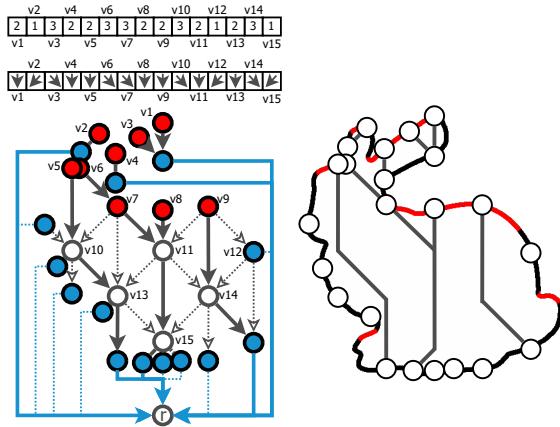


Figure 6: L'encodage par aiguillage (à gauche) et les supports 2D résultants (à droite).

c'est l'arête dirigée vers la droite qui est activée. Cette méthode d'encodage ne peut encoder que les sous-arbres d'un graphe car pour chaque sommet v_i , une seule arête sortante dans $\text{out}(v_i)$ peut être activée. L'ensemble des solutions potentielles est donc l'ensemble des sous-arbres contenus dans le graphe initial enracinés en r . Puisque la solution du DSTP doit être un arbre enraciné en r , l'*encodage par aiguillage* est particulièrement adapté pour résoudre le problème LS²DO.

L'exécution de l'algorithme génétique est décrite par le logigramme de la figure 7. Afin de s'adapter à la résolution du problème LS²DO, l'algorithme génétique se paramètre par le choix de 4 opérateurs pour la sélection, le crossover, la mutation et la réinsertion, ainsi que 4 paramètres d'exécution qui influencent l'algorithme : la taille minimale d'une génération (MinSize), la variation de taille d'une génération (en ajoutant l'envergure à la taille minimale on obtient la taille maximale d'une génération), la probabilité de Crossover (CP) et la probabilité de Mutation (MP). Le choix de ces paramètres a été réalisé au travers d'un plan d'expérience qui n'est pas détaillé dans cet article. Les opérateurs sélectionnés sont en revanche listés dans le tableau 1

Paramètres	Niveaux optimisés
Sélection	Tournament
Crossover	Uniform
Mutation	Twors
Réinsertion	Elitist
Taille min. de génération	500
Variation de taille de génération	500
Probabilité de Crossover	0.75
Probabilité de Mutation	0.3

Table 1: Paramètres optimisés de l'algorithme génétique implémentés dans la méthodologie proposée.

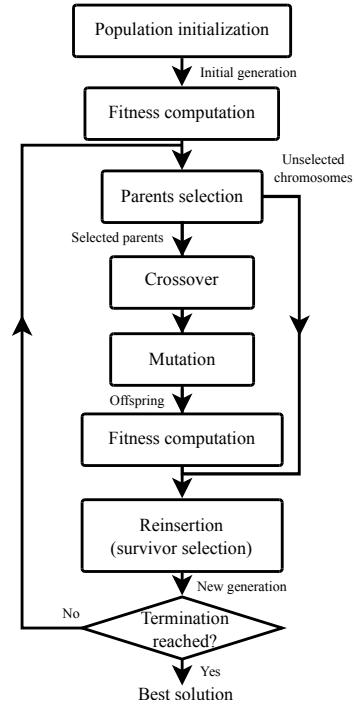


Figure 7: Logigramme d'exécution d'un algorithme génétique

5. Comparaison des résultats

Pour évaluer l'intérêt de l'optimisation de lattices pour supporter les zones en contre-dépouille, une étude approfondie a été réalisée comparant les supports générés par notre approche aux supports proposés dans les travaux de Schmidt et al., implémentés dans le logiciel MeshMixer [SU14] et ceux générés par un autre logiciel classiquement utilisé dans l'industrie. Pour ce dernier, deux stratégies sont comparées à la méthodologie proposée : une dédiée à la génération de structures supports pour la technologie SLA, et l'autre pour la technologie SLM. Cette comparaison est faite pour les supports intérieurs et extérieurs de trois cas tests (voir figure 8, 10 et 9). Pour détacher les pièces du plateau d'impression, celles-ci sont positionnées à une hauteur constante afin de permettre le passage d'un outil de découpe. Il doit être noté que pour cette comparaison, les supports Meshmixer et les supports SLA ont été générés avec le même diamètre que ceux générés par la méthodologie proposée ($d = 0.5\text{mm}$). En plus de ces trois cas tests, deux autres modèles fournis par Vanek et al. [VGB14] ont été supportés avec à notre méthodologie, avec Meshmixer, et avec les stratégies SLA et SLM, afin d'avoir une comparaison indépendante : l'Armadillo de Stanford, et un modèle d'oiseau. Il doit être noté que la distance de contre-dépouille maximale o_p utilisée par Vanek et al. est plus grande que celle utilisée dans

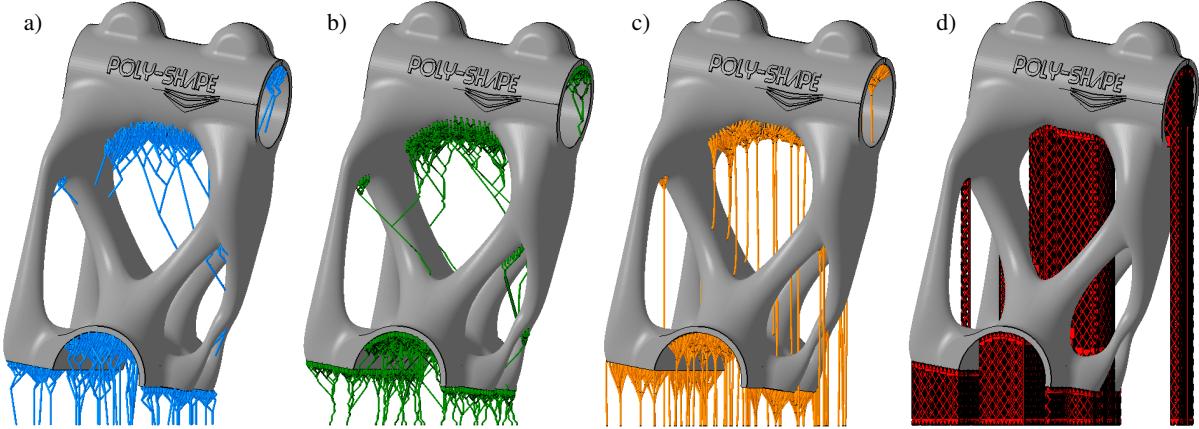


Figure 8: Supports extérieurs de la Potence générés par notre méthodologie (a), par Meshmixer [SU14] (b), par une stratégie de supports SLA (c) et par une stratégie de supports SLM (d).

	Potence internes	Potence externes	Turbine internes	Turbine externes	Lapin de Stanford internes	Lapin de Stanford externes	Armadillo externes	Oiseau externes
Volumes (en mm ³)								
LS ² DO par algorithme génétique	420	1300	45	3130	210	310	740	810
Schmidt et al. [SU14], Meshmixer	460	1450	85	3450	245	390	750	1000
stratégie SLA	565	1850	70	6990	300	400	940	2280
stratégie SLM	630	3100	90	5760	380	470	1370	3650
Vanek et al. [VGB14]	N/A	N/A	N/A	N/A	N/A	N/A	30*	20*
Réductions de volume								
LS ² DO par rapport à Meshmixer	-9%	-10%	-47%	-9%	-14%	-21%	-1%	-23%
LS ² DO par rapport à la stratégie SLA	-26%	-30%	-36%	-55%	-30%	-23%	-27%	-181%
LS ² DO par rapport à la stratégie SLM	-33%	-58%	-50%	-46%	-45%	-34%	-85%	-351%

* avec des murs de 10µm d'épaisseur

Table 2: Comparaison des volumes de supports générés par notre méthodologie et par les autres stratégies.

	Potence	Turbine	Lapin de Stanford	Armadillo	Oiseau
Temps de calcul					
LS ² DO par algorithme génétique	75min	48min	26min	33min	18min
Schmidt et al. [SU14], Meshmixer	30min	154min	21min	76min	83min
stratégie SLA	3min 21s	25s	11s	9s	2s
stratégie SLM	1min 10s	3min 5s	20s	11s	4s
Vanek et al. [VGB14]	N/A	N/A	N/A	161min	80min

Table 3: Comparaison des temps de calcul de la méthodologie proposée et des autres stratégies considérées, sur les supports externes des différents cas tests.

cet article, puisque la technologie de FA considérée est différente (LBM dans cet article et FDM dans leur cas).

5.1. Comparaison des volumes de supports

Tout d'abord, avec la stratégie Meshmixer, il peut être remarqué que certains supports contournent la pièce (sous le menton du Lapin de Stanford notamment), afin d'éviter de rattacher la base du support à la pièce. De la même façon, avec la stratégie de supports SLA, certains piliers supports

sont directement connectés au plateau de fabrication, alors qu'une connexion inclinée directement à la pièce permettrait de réduire leur volume. L'objectif est probablement d'éviter les opérations de finition additionnelles, au niveau des surfaces de jonction, qui peuvent être chronophages. Cependant, la consommation de matière, et donc le temps de production sont grandement augmentés par ce choix.

Il peut également être observé que les supports générés par la stratégie de Meshmixer ne sont pas rectilignes (et présentent une forme anguleuse), contrairement aux sup-

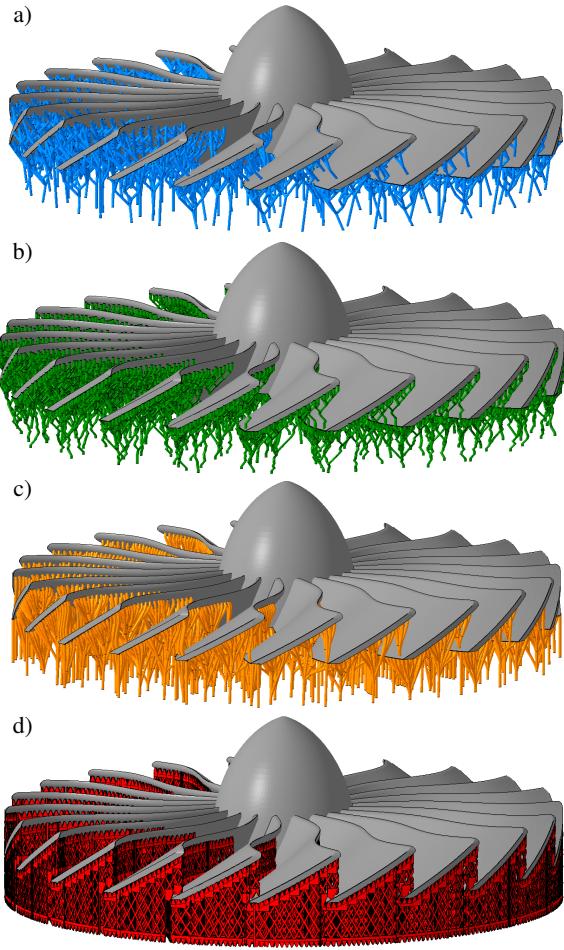


Figure 9: Supports extérieurs de la Turbine générés par notre méthodologie (a), par Meshmixer [SU14] (b), par une stratégie de supports SLA (c) et par une stratégie de supports SLM (d).

ports générés par la méthodologie proposée. Cela est probablement dû à la stratégie d'optimisation exécutée et entraîne une dégradation de la résistance mécanique des supports. Contrairement à la méthodologie proposée et à la stratégie de Meshmixer, la stratégie SLA présente des supports en forme d'arbres avec des troncs verticaux : près des surfaces en contre-dépouille, les poutres sont fortement groupées mais au-delà d'une certaine distance, il semble qu'aucune autre réunion de poutres ne soit réalisable, et les supports finissent avec un pilier vertical. Ils sont donc moins optimisés que ceux générés par la méthodologie proposée ou la stratégie Meshmixer, et les volumes résultants sont plus importants.

La stratégie SLM est mentionnée ici car c'est la principale stratégie de supportage utilisée pour la technolo-

gie LBM. Elle est constituée de fins murs verticaux, possédant des perforations en forme de losange. Les supports générés sont globalement plus denses que les supports en forme d'arbres, car ils permettent également de supporter les contraintes résiduelles de la pièce. Ils sont aussi plus difficiles à retirer que les supports générés par l'approche LS²DO proposée. Cependant, la comparaison des deux donne une idée de l'intervalle d'optimisation pour le volume des supports : le meilleur compromis entre des supports denses permettant de minimiser les déformations de la pièce, et des supports légers soutenant uniquement les contre-dépouilles, devrait se trouver entre les supports générés par ces deux stratégies.

Plus précisément, le tableau 2 compare le volume des supports obtenus en utilisant la méthodologie proposée, et le volume des supports générés par Meshmixer et par les stratégies SLA et SLM. Il apparaît clairement que la méthodologie proposée permet la génération des structures supports avec les volumes les plus faibles. Le tableau 2 indique également le volume des supports générés par Vanek et al. [VGB14] pour l'Armadillo et le modèle d'oiseau seulement, le code de Vanek et al. n'étant pas disponible (N/A) pour les tester sur les autres exemples. Cependant, ces structures présentent des épaisseurs de murs d'approximativement 10 μm sur toute leur hauteur, ce qui est techniquement impossible à fabriquer en LBM ou en FDM pour le moment. Ainsi, ces volumes ne peuvent pas équitablement être comparés à ceux des supports obtenus en utilisant notre méthodologie, et ne sont mentionnés qu'à titre indicatif.

5.2. Comparaison du temps de calcul

Une comparaison du temps de calcul nécessaire à la génération des supports précédemment présentés est détaillée dans le tableau 3. Pour le Lapin de Stanford, la Potence et la Turbine, ces résultats incluent le temps de génération des structures supports internes et externes. Les résultats du LS²DO, de Meshmixer et des stratégies SLA et SLM ont été obtenues à en utilisant un processeur Intel® Core™ i7-4710HQ CPU à 2.50GHz, avec 16GB de RAM, et une carte GPU Nvidia GeForce GTX 970M. Ces caractéristiques sont similaires à celles utilisées par Vanek et al. [VGB14] pour obtenir leurs structures supports (mentionnées dans les dernières lignes de la table 3). Ainsi, les temps de calcul peuvent être équitablement comparés entre eux. Il est important de mentionner qu'avec l'implémentation actuelle de notre méthodologie d'optimisation, aucun calcul n'est parallélisé sur carte graphique (GPU).

Ces résultats démontrent que les logiciels utilisés par l'industrie pour générer des structures supports sont les plus rapides (à savoir les stratégies SLA et SLM). En plus de mettre probablement en œuvre la parallélisation sur carte graphique, ces écarts de temps peuvent s'expliquer par le fait que les structures résultantes sont moins optimisées optimisées que celles générées par les autres méthodes (comme

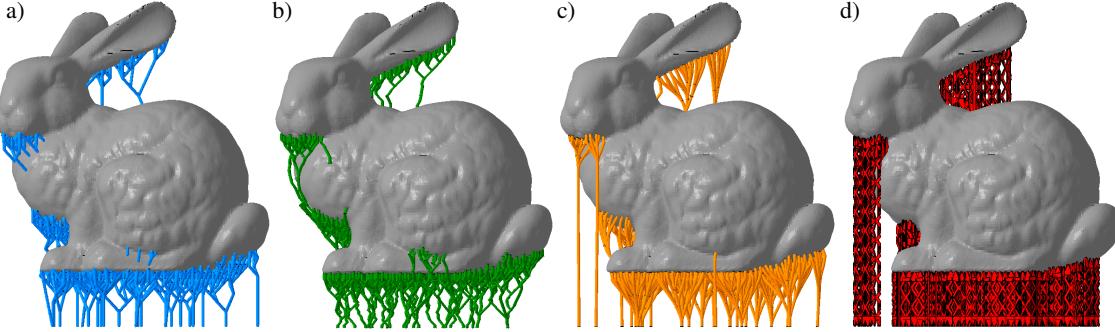


Figure 10: Supports extérieurs du Lapin de Stanford générés par notre méthodologie (a), par Meshmixer [SU14] (b), par une stratégie de supports SLA (c) et par une stratégie de supports SLM (d).

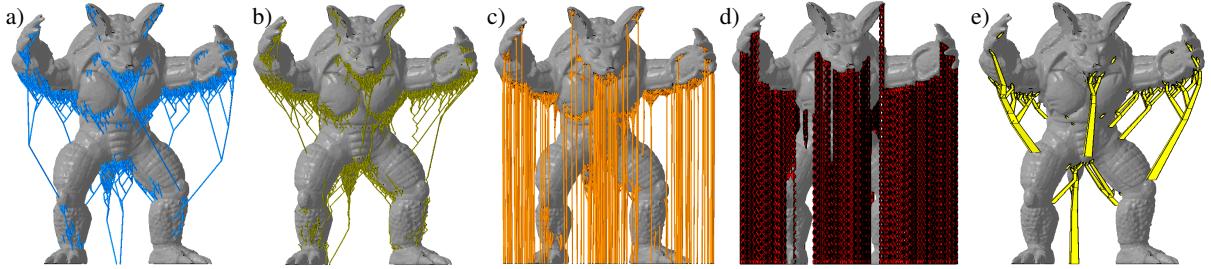


Figure 11: Supports extérieurs de l'Armadillo de Stanford générés par notre méthodologie (a), par Meshmixer [SU14] (b), par une stratégie de supports SLA (c) par une stratégie de supports SLM (d) et par Vanek et al. (e).

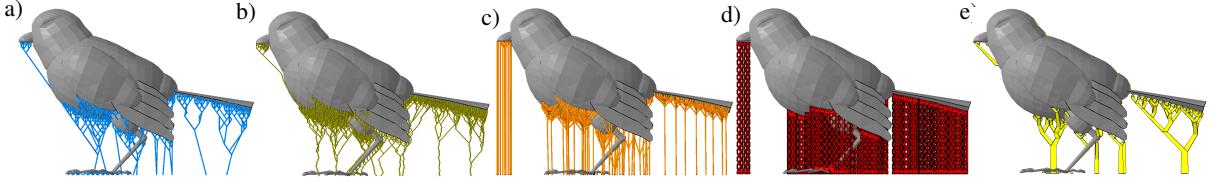


Figure 12: Supports extérieurs du modèle d'oiseau générés par notre méthodologie (a), par Meshmixer [SU14] (b), par une stratégie de supports SLA (c) par une stratégie de supports SLM (d) et par Vanek et al. (e).

illustré par les valeurs de la table 2). Cependant, il peut être noté que la méthodologie proposée est plus rapide que les travaux de Schmidt et al. [SU14] implémentés dans Meshmixer sur certains cas tests. Ceci pourrait s'expliquer par le fait que leurs structures supports contournent la pièce par endroits (comme détaillé dans la section 5.1) : imposer de telles contraintes nécessite plus de calcul et donc des temps d'optimisation plus importants.

En comparaison des travaux de Vanek et al. [VGB14], la méthodologie présentée dans cet article est plus rapide. Cela peut être dû à la différence de nature des deux problèmes considérés, celui identifié dans cet article étant discret, alors que celui résolu par Vanek et al. est continu. Par ailleurs, la distance de contre-dépouille maximale utilisée dans leurs travaux est plus importante que celle utilisée ici, du fait que

les technologies de FA diffèrent (LBM versus FDM). Si ils utilisaient la même distance de contre-dépouille maximale que celle utilisée dans cet article, le nombre de points de connection sur la surface en contre-dépouille serait plus important, et leur optimisation prendrait plus de temps à s'exécuter. La comparaison du tableau 3 n'est donc pas complètement équitable, à notre défaveur.

En conclusion, en plus de générer des structures supports avec moins de volume, la méthodologie proposée présente également des temps de calcul réduits par rapport à d'autres travaux de pointe identifiés dans la littérature. Cependant, cela ne rivalise pas encore avec les logiciels présents sur le marché, exploitant la parallélisation GPU. Néanmoins, en plus d'implémenter le calcul par GPU, la méthodologie proposée pourrait être optimisée en couplant l'algorithme gé-



Figure 13: La Potence imprimée en LBM et coupée en deux afin de faire apparaître les supports internes.

nétique avec un algorithme de recherche heuristique, ou en créant un opérateur génétique spécifique (pour le crossover ou la mutation par exemple), plus adapté au problème particulier du LS²DO.

6. Conclusions et perspectives

Une nouvelle méthodologie a été proposée afin d'optimiser les supports pour les procédés de FA. L'objectif de cette méthodologie est de soutenir toutes les zones en contre-dépouille d'une pièce, laissant pour le moment de côté les problématiques de déformations et d'accumulations thermiques. Pour cela, un lattice fabricable préalablement optimisé est généré sous les surfaces en contre-dépouille. Ensuite, un algorithme Génétique optimise ce lattice en lui retirant le maximum de poutres, tout en assurant que toutes les zones à supporter sont encore soutenues. Naturellement, travailler avec une structure en forme d'arbre telle que celle générée facilite également le retrait des supports extérieurs lors des phases de finition.

Divers résultats ont été présentés : les structures supports intérieures et extérieures de 5 cas tests ont été générées et imprimées avec succès, et leurs volumes ont été comparés à ceux des supports générés par plusieurs stratégies de pointe dans le domaine, soulignant l'intérêt de l'algorithme développé en terme d'optimisation de volume. Au travers des résultats comparés, il a été démontré qu'utiliser un algorithme génétique pour résoudre le problème LS²DO permet de générer des supports plus optimisés que les stratégies traditionnelles de génération de supports. Les temps de calcul ont également été comparés et si les logiciels commerciaux génèrent des structures avec des volumes moins optimisés mais de manière plus rapide, la méthodologie proposée est plus rapide que les autres approches académiques.

Comme perspective pour cette recherche, la conver-

gence de l'algorithme génétique pourrait être améliorée. Par exemple, une recherche heuristique pourrait être implantée afin de générer de meilleures populations initiales. Une recherche locale rapide pourrait également être effectuée après chaque crossover ou mutation, afin d'obtenir de meilleurs chromosomes enfants. De plus, afin de réduire le temps d'optimisation de l'algorithme génétique, les diverses zones en contre-dépouille pourraient être optimisées indépendamment. Chaque groupe contiendrait alors moins de surfaces en contre-dépouille et donc moins de poutres dans le lattice initial. Ainsi, leur optimisation convergerait plus rapidement, résultant en un temps d'optimisation global plus court. Naturellement, une telle décomposition pourrait également bénéficier de l'implémentation de la parallélisation GPU.

Afin d'améliorer plus en profondeur l'optimisation de structures supports, la fonction objectif (qui ne comprend que le volume de matière) pourrait être étendue, en prenant en compte le retrait des supports et les coûts de finition. Cependant, ces coûts peuvent être difficiles à estimer car ils dépendent de nombreux facteurs qualitatifs (comme par exemple les outils utilisés ou encore l'expérience de l'opérateur de finition).

Enfin, du fait que les déformations et les problèmes d'accumulation thermique ont été laissés de côté, la méthodologie proposée n'est qu'un premier bloc dans le large domaine de l'optimisation de supports en FA. Son couplage avec des algorithmes d'optimisation thermomécanique est d'intérêt dans le futur, afin de générer des supports polyfonctionnels, qui peuvent soutenir les contre-dépouilles, rigidifier les géométries sujettes aux déformations, et dissiper les zones d'accumulation thermique de n'importe quelle pièce produite en FA.

Références

- [BB16] BROOKS H., BRIGDEN K. : Design of conformal cooling layers with self-supporting lattices for additively manufactured tooling. *Additive Manufacturing*. Vol. 11 (juillet 2016), 16–22.
- [Boy15] BOYARD N. : *Méthodologie de conception pour la réalisation de pièces en Fabrication Additive*. PhD thesis, 2015.
- [Cal14] CALIGNANO F. : Design optimization of supports for overhanging structures in aluminum and titanium alloys by selective laser melting. *Materials & Design*. Vol. 64 (décembre 2014), 203–213.
- [CC15] CHENG B., CHOU K. : Geometric consideration of support structures in part overhang fabrications by electron beam additive manufacturing. *Computer-Aided Design*. Vol. 69 (décembre 2015), 102–111.
- [CCC*00] CHARIKAR M., CHEKURI C., CHEUNG T., DAI Z., GOEL A., GUHA S., LI M. : Approximation

- Algorithms for Directed Steiner Problems. *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms* (2000), 15.
- [CPVA17] CHOURANI L., PERNOT J.-P., VÉRON P., ABED S. : Lattice structure lightweight triangulation for additive manufacturing. *Computer-Aided Design*. Vol. 90 (septembre 2017), 95–104.
- [CSCC15] COOPER K., STEELE P., CHENG B., CHOU K. : Contact-Free Support Structures for Part Overhangs in Powder-Bed Metal Additive Manufacturing. 12.
- [CSW13] CLOOTS M., SPIERINGS A. B., WEGENER K. : Assessing new support minimizing strategies for the additive manufacturing technology SLM. In *24th International SFF Symposium-An Additive Manufacturing Conference, Austin, USA, University of Texas at Austin* (2013), pp. 631–643.
- [DGYD17] DILBEROGLU U. M., GHAREHPAPAGH B., YAMAN U., DOLEN M. : The Role of Additive Manufacturing in the Era of Industry 4.0. *Procedia Manufacturing*. Vol. 11 (2017), 545–554.
- [DHL14] DUMAS J., HERGEL J., LEFEBVRE S. : Bridging the Gap : Automated Steady Scaffoldings for 3d Printing. *ACM Transactions on Graphics (TOG)*. Vol. 33, Num. 4 (2014), 10.
- [Gia18] GIACOMELLI D. : GeneticSharp, 2018.
- [GW16] GAN M., WONG C. : Practical support structures for selective laser melting. *Journal of Materials Processing Technology*. Vol. 238 (décembre 2016), 474–484.
- [HYH*13] HUSSEIN A., HAO L., YAN C., EVERSON R., YOUNG P. : Advanced lattice support structures for metal additive manufacturing. *Journal of Materials Processing Technology*. Vol. 213, Num. 7 (juillet 2013), 1019–1026.
- [HK03] HALPERIN E., KRAUTHGAMER R. : Polylogarithmic inapproximability. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing* (2003), ACM, pp. 585–594.
- [HK13] HAUPTMANN M., KARPIŃSKI M. : *A compendium on Steiner tree problems*. Inst. für Informatik, 2013.
- [Hus13] HUSSEIN A. : *The Development of Lightweight Cellular Structures for Metal Additive Manufacturing*. PhD thesis, 2013.
- [HYML09] HUANG X., YE C., MO J., LIU H. : Slice data based support generation algorithm for fused deposition modeling. *Tsinghua Science & Technology*. Vol. 14 (2009), 223–228.
- [JBAG12] JHABVALA J., BOILLAT E., ANDRÉ C., GLARDON R. : An innovative method to build support structures with a pulsed laser in the selective laser melting process. *The International Journal of Advanced Manufacturing Technology*. Vol. 59, Num. 1-4 (mars 2012), 137–142.
- [JHF15] JIN Y.-A., HE Y., FU J.-Z. : Support generation for additive manufacturing based on sliced data. *The International Journal of Advanced Manufacturing Technology*. Vol. 80, Num. 9-12 (octobre 2015), 2041–2052.
- [JML*14] JÄRVINEN J.-P., MATILAINEN V., LI X., PIILI H., SALMINEN A., MÄKELÄ I., NYRHILÄ O. : Characterization of Effect of Support Structures in Laser Additive Manufacturing of Stainless Steel. *Physics Procedia*. Vol. 56 (2014), 72–81.
- [KZS12] KROL T. A., ZAEH M. F., SEIDEL C. : Optimization of Supports in Metal-Based Additive Manufacturing by Means of Finite Element Models. 12.
- [Lan17] LANGELAAR M. : An additive manufacturing filter for topology optimization of print-ready designs. *Structural and Multidisciplinary Optimization*. Vol. 55, Num. 3 (mars 2017), 871–883.
- [LDJ*15] LI D., DAI N., JIANG X., SHEN Z., CHEN X. : Density Aware Internal Supporting Structure Modeling of 3d Printed Objects. IEEE, pp. 209–215.
- [LL16] LEE J., LEE K. : Block-based inner support structure generation algorithm for 3d printing using fused deposition modeling. *The International Journal of Advanced Manufacturing Technology* (2016), 2151–2163.
- [noa12] ASTM F2792-12a, Standard Terminology for Additive Manufacturing Technologies, (Withdrawn 2015), 2012.
- [PAHA18] PANESAR A., ABDI M., HICKMAN D., ASHCROFT I. : Strategies for functionally graded lattice structures derived using topology optimisation for Additive Manufacturing. *Additive Manufacturing*. Vol. 19 (janvier 2018), 81–94.
- [QLYG12] QIAN B., LICHAO Z., YUSHENG S., GUOCHENG L. : Support fast generation algorithm Based on discrete-marking in rapid prototyping. *Affective Computing and Intelligent Interaction* (2012), 683–695.
- [SU14] SCHMIDT R., UMETANI N. : Branching support structures for 3d printing. In *ACM SIGGRAPH 2014 Studio* (2014), ACM, p. 9.
- [Tho09] THOMAS D. : *The Development of Design Rules for Selective Laser Melting*. PhD thesis, 2009.
- [VA16] VAIDYA R., ANAND S. : Optimum Support Structure Generation for Additive Manufacturing Using Unit Cell Structures and Support Removal Constraint. *Procedia Manufacturing*. Vol. 5 (2016), 1043–1059.
- [VGB14] VANEK J., GALICIA J. A. G., BENES B. : Clever Support : Efficient Support Structure Generation for Digital Fabrication. *Computer Graphics Forum*. Vol. 33, Num. 5 (août 2014), 117–125.
- [Zel97] ZELIKOVSKY A. : A series of approximation algorithms for the acyclic directed Steiner tree problem. *Algorithmica*. Vol. 18, Num. 1 (1997), 99–110.