Rafael Pereira Ferreira

AN ARTIFICIALLY INTELLIGENT SPACE-FILLING TRAJECTORY PLANNING FOR WIRE ARC ADDITIVE MANUFACTURING

UFU

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA MECÂNICA
2023

Rafael Pereira Ferreira

AN ARTIFICIALLY INTELLIGENT SPACE-FILLING TRAJECTORY PLANNING FOR WIRE ARC ADDITIVE MANUFACTURING

Thesis submitted to the Post-Graduation Program in Mechanical Engineering of the Federal University of Uberlândia as part of the requirements for obtaining the degree in **DOCTOR OF MECHANICAL ENGINEERING.**

Concentration area: Materials and Manufacturing Processes

Supervisor: Prof. Dr. Américo Scotti

UBERLÂNDIA - MG

2023

Ficha Catalográfica Online do Sistema de Bibliotecas da UFU
com dados informados pelo(a) próprio(a) autor(a).

F383
2023

Ferreira, Rafael Pereira, 1987-
AN ARTIFICIALLY INTELLIGENT SPACE-FILLING TRAJECTORY PLANNING FOR WIRE ARC ADDITIVE MANUFACTURING [recurso eletrônico] / Rafael Pereira Ferreira. - 2023.

Orientador: Américo Scotti.
Tese (Doutorado) - Universidade Federal de Uberlândia, Pós-graduação em Engenharia Mecânica.
Modo de acesso: Internet.
Disponível em: http://doi.org/10.14393/ufu.te.2023.431
Inclui bibliografia.
Inclui ilustrações.

1. Engenharia mecânica. I. Scotti, Américo ,1955-, (Orient.). II. Universidade Federal de Uberlândia. Pós-graduação em Engenharia Mecânica. III. Título.

CDU: 621

Bibliotecários responsáveis pela estrutura de acordo com o AACR2:
Gizele Cristine Nunes do Couto - CRB6/2091
Nelson Marcos Ferreira - CRB6/3074

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
Coordenação do Programa de Pós-Graduação em Engenharia Mecânica
Av. João Naves de Ávila, nº 2121, Bloco 1M, Sala 212 - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902
Telefone: (34) 3239-4282 - www.posmecanicaufu.com.br - secposmec@mecanica.ufu.br

ATA DE DEFESA - PÓS-GRADUAÇÃO

Programa de Pós-Graduação em:	Engenharia Mecânica				
Defesa de:	Tese de Doutorado Acadêmico, nº 363, PPGEM				
Data:	08/08/2023	Hora de início:	09:00	Hora de encerramento:	12:30
Matrícula do Discente:	11923EMC009				
Nome do Discente:	Rafael Pereira Ferreira				
Título do Trabalho:	An Artificially Intelligent Space-Filling Trajectory Planning for Wire Arc Additive Manufacturing				
Área de concentração:	Materiais e Processos de Fabricação				
Linha de pesquisa:	Processos de Fabricação (Usinagem e Soldagem)				
Projeto de Pesquisa de vinculação:	CAPES-PRINT (88887.696939/2022-00), bolsa para realização do doutorado sanduíche na Alemanha em colaboração com o Welding and Joining Institute (ISF) of RWTH-Aachen University/Abicor Binzel; Special Diffuser for GMAW torches applied in WAAM: an exploratory Research plan to prove the concept", acordado sob o Termo de Cooperação Técnica nº 23117.002616/2022-27, estabelecido entre a Binzel Abicor, Alemanha, a Universidade Federal de Uberlândia (UFU), a Fundação de Apoio Universitário (FAU), e a Embrapii Unidade FEMEC; Performance Evaluation of New Shielding Gases in Wire-Arc Additive Manufacturing", acordado sob o Termo de Cooperação Técnica nº 23117.051893/2021-82, estabelecido entre a empresa LINDE GmbH, Linde Gas (LINDE), Alemanha, a Universidade Federal de Uberlândia (UFU) e a Fundação de Apoio Universitário (FAU)				

Reuniu-se por meio híbrido (presencial e por videoconferência) a Banca Examinadora, designada pelo Colegiado do Programa de Pós-graduação em Engenharia Mecânica, assim composta: Professores Doutores: Marcus Antônio Viana Duarte - FEMEC/UFU; Arthur Alves Fiocchi - FEMEC/UFU; Rodrigo Minetto - UTFPR; João Pedro de Sousa Oliveira - UNL (Portugal); Emil Schubert - Abicor Binzel (Alemanha) e Américo Scotti - FEMEC/UFU, orientador do candidato.

Iniciando os trabalhos, o presidente da mesa, Dr. Américo Scotti, apresentou a Comissão Examinadora e o candidato, agradeceu a presença do público, e concedeu ao Discente a palavra para a exposição do seu trabalho. A duração da apresentação do Discente e o tempo de arguição e resposta foram conforme as normas do Programa.

A seguir o senhor(a) presidente concedeu a palavra, pela ordem sucessivamente, aos(às) examinadores(as), que passaram a arguir o(a) candidato(a). Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando o(a) candidato(a):

Aprovado.

Esta defesa faz parte dos requisitos necessários à obtenção do título de Doutor.

O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação interna da UFU.

Nada mais havendo a tratar foram encerrados os trabalhos. Foi lavrada a presente ata que após lida e achada conforme foi assinada pela Banca Examinadora.

Documento assinado eletronicamente por **Américo Scotti**, **Professor(a) do Magistério Superior**, em 08/08/2023, às 14:58, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **João Pedro de Sousa Oliveira**, **Usuário Externo**, em 08/08/2023, às 16:10, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Rodrigo Minetto**, **Usuário Externo**, em 08/08/2023, às 16:25, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Emil Schubert**, **Usuário Externo**, em 08/08/2023, às 21:14, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Marcus Antonio Viana Duarte**, **Professor(a) do Magistério Superior**, em 08/08/2023, às 22:19, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Arthur Alves Fiocchi**, **Professor(a) do Magistério Superior**, em 09/08/2023, às 08:17, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **4718974** e o código CRC **06E4721A**.

Dedico,

Aos meus pais.

A minha esposa, Mary, e meu filho, Rael.

E em memória ao meu cachorro e companheiro, Vegê.

AGRADECIMENTOS
(Acknowledgements)

Ao professor Américo Scotti, pelos diálogos, orientações, conselhos e incentivos ao longo dos anos de doutorado. Sempre tentei ver toda conversa como uma oportunidade de crescimento e melhoria para meu trabalho. Sempre tentei agarrar as oportunidades que foram abertas durante esses anos, tais como, ministrar palestras, participar de conferência nacionais e internacionais, realizar visitas técnicas e aplicar minha pesquisa em um centro de pesquisa de uma empresa internacional. Gratidão.

Aos professores Louriel Vilarinho, Ruham Reis, Fran Sérgio, Duarte e Valtair Ferraresi, pelo conhecimento repassado durante as disciplinas.

Aos professore Luiz Paes e Leandro João pelo incentivo e ajuda em minha pesquisa e troca de experiências.

Aos engenheiros, técnicos e alunos de iniciação científica do Laprosolda/UFU, pela ajuda e pelos momentos de descontração compartilhados. Em especial aos Henrique Ferraresi e Edmundo Benedetti pela ajuda durante meus primeiros experimentos práticos com a fonte de soldagem.

Aos meus colegas de pós-graduação Fernando Scotti, Vinicius Lemes, Felipe Teixeira e Rodrigo Farias pelo apoio durante a realização de experimentos, discussões, aprendizado e, principalmente, pela amizade.

Agradeço sinceramente à Abicor Binzel, em especial ao Prof. Emil Schubert, pela inestimável oportunidade de realizar parte desta pesquisa no Innovation Technology Center (ITC) da empresa.Aos membros da banca, que doaram tempo e experiência para aperfeiçoar esta tese.

Ao IFMA por ceder meu afastamento por 4 anos para conclusão deste doutorado (Portaria N° 5.356, de 27 de agosto de 2019), à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (88887.696939/2022-00) pela bolsa para realização do doutorado sanduíche na Alemanha em colaboração com o Welding and Joining Institute (ISF) of RWTH-Aachen University/Abicor Binzel e à PETROBRAS (Projeto SEI nº 23117.018175/2019-80) pelo material cedido para meus primeiros experimentos.

Aos meus pais, por todo amor, carinho e compreensão diante da distância mantida durante a realização deste trabalho.

A minha esposa que comprou a ideia de me acompanhar durante esses 4 anos de doutorado. Nesses anos tivemos muitos momentos bons e desafios que nos fizeram mais fortes,

passamos juntos por pandemia, a morte de nosso querido pet Vegê, o nascimento do nosso amado filho Rael e a missão de viver no exterior. Evoluímos de fato e sem você nada disso seria possível.

FERREIRA, R. P. **An Artificially Intelligent Space-Filling Trajectory Planning for Wire Arc Additive Manufacturing. 2023.** 159 p. PhD Thesis, Federal University of Uberlândia, Uberlândia, MG, Brazil

ABSTRACT

This thesis systematically explored the implementation of a Space-Filling strategy for an artificially intelligent trajectory planning to be used in Wire Arc Additive Manufacturing (WAAM) and investigated its benefits and challenges. The Pixel strategy, as the focus, was proposed and developed as an innovative and flexible computerized tool in trajectory planning for complex geometries. Pixel was intended to provide multiple applicable trajectories for part printings and the subsequent optimized trajectory selection for each case. To achieve this target, a basic version was offered using a space-filling approach, by formulating a grid of nodes, and, simultaneously, four heuristics for node connections. Computational evaluations demonstrated the effectiveness of the "Basic-Pixel" strategy for various part geometries. Experimental builds using Gas Metal Arc (GMA) and plain carbon steel confirmed the practical viability of this basic version, enabling the deposition and construction of intricate shapes, including polygonal nonconvex geometries with holes. To boost the algorithm's performance, the "Enhanced-Pixel" strategy was introduced, incorporating a new node sorting method and four trajectory planning heuristics. Comparative analyses in specific case studies validated the operational efficiency and effectiveness of the "enhanced" version compared to commercially applied conventional strategies. The study further explored the following "Advanced-Pixel" strategy, utilizing reinforcement learning techniques (artificial intelligence) to optimize the selection of trajectory planning heuristics and ordering methods. Experimental analyses revealed that the "Advanced-Pixel" strategy outperforms the "Enhanced-Pixel" strategy in terms of performance gains and response quality, demonstrating reduced printing time and trajectory distance, particularly for larger components. Additionally, the thesis work investigated the "Fast-Pixel" strategy, leveraging clustering techniques with "k-means" to reduce the dimensionality of the optimization problem. The "Fast-Pixel" strategy implementation demonstrated improved performance across all tested parts, significantly reducing computational time while improving response quality. At last, the thesis text outlines future research directions, including expanding to different materials, optimizing computational efficiency, mitigating non-conformities, exploring hybrid strategies, and developing real-time monitoring and quality control systems. In conclusion, the research and development work in this thesis, by introducing the Pixel strategy and its improvements, provided an option for trajectory planning in WAAM. The experimental validations, computational evaluations, and practical demonstrations highlighted the effectiveness and viability of the proposed strategies. These scientific-oriented developments have significant implications for the efficient and effective production of complex parts using additive manufacturing technologies, paving the way for further advancements in the field.

Keywords: Wire Arc Additive Manufacturing (WAAM); Trajectory planning; Space-Filling strategy; Travelling Salesman Problem; Operational efficiency and effectiveness; Reinforcement learning, Multi-Armed Bandit; Clustering; K-means.

FERREIRA, R. P. **Um Planejamento Artificialmente Inteligente de Trajetórias por Preenchimento de Espaços para Manufatura Aditiva por Deposição a Arco (MADA). 2023.** 159 f. Tese de Doutorado, Universidade Federal de Uberlândia, Uberlândia, Minas Gerais, Brasil

RESUMO

Esta tese explorou sistematicamente a implementação de uma estratégia de preenchimento de espaço para um planejamento de trajetória artificialmente inteligente para ser usado em na Manufatura Aditiva por Deposição a Arco (MADA) e investigou seus benefícios e desafios. A estratégia Pixel, como foco, foi proposta e desenvolvida como uma ferramenta informatizada inovadora e flexível no planejamento de trajetórias para geometrias complexas. O Pixel foi projetado para fornecer múltiplas trajetórias aplicáveis para impressões de peças e a subseqüente seleção de trajetória otimizada para cada caso. Para atingir esse objetivo, uma versão básica foi oferecida usando uma abordagem de preenchimento de espaço, formulando uma grade de nós e, simultaneamente, quatro heurísticas para conexões de nós. Avaliações computacionais demonstraram a eficácia da estratégia "Basic-Pixel" para várias geometrias de peças. Construções experimentais usando Gas Metal Arc (GMA) e aço comum ao carbono confirmaram a viabilidade prática desta versão básica, permitindo a deposição e construção de formas intrincadas, incluindo geometrias poligonais não convexas com furos. Para aumentar o desempenho do algoritmo, a estratégia "Enhanced-Pixel" foi introduzida, incorporando um novo método de classificação de nós e quatro heurísticas de planejamento de trajetória. Análises comparativas em estudos de caso específicos validaram a eficiência operacional e eficácia da versão "aprimorada" em comparação com estratégias convencionais aplicadas comercialmente. O estudo explorou ainda mais a seguinte a estratégia "Advanced-Pixel", utilizando técnicas de aprendizado por reforço (inteligência artificial) para otimizar a seleção de heurísticas de planejamento de trajetória e métodos de ordenação. Análises experimentais revelaram que a estratégia "Advanced-Pixel" supera a estratégia "Enhanced-Pixel" em termos de ganhos de desempenho e qualidade de resposta, demonstrando tempo de impressão e distância de trajetória reduzidos, principalmente para componentes maiores. Adicionalmente, o trabalho de tese investigou a estratégia "Fast-Pixel", aproveitando técnicas de agrupamento com "k-means" para reduzir a dimensionalidade do problema de otimização. A implementação da estratégia "Fast-Pixel" demonstrou desempenho aprimorado em todas as peças testadas, reduzindo significativamente o tempo computacional e melhorando a qualidade da resposta. Por fim, o texto da tese delineia direções futuras de pesquisa, incluindo a expansão para diferentes materiais, otimização da eficiência computacional, mitigação de não conformidades, exploração de estratégias híbridas e desenvolvimento de sistemas de monitoramento e controle de qualidade em tempo real. Em conclusão, o trabalho de pesquisa e desenvolvimento desta tese, ao apresentar a estratégia Pixel e suas melhorias, forneceu uma opção de planejamento de trajetória para MADA. As validações experimentais, avaliações computacionais e demonstrações práticas evidenciaram a eficácia e viabilidade das estratégias propostas. Esses desenvolvimentos de orientação científica têm implicações significativas para a produção eficiente e eficaz de peças complexas usando tecnologias de manufatura aditiva, abrindo caminho para novos avanços no campo.

Palavras-Chave: Manufatura Aditiva por Deposição a Arco (MADA); Planejamento de trajetória; Estratégia de Preenchimento de Espaço; Problema do Caixeiro Viajante; Eficiência e efetividade operacional; Aprendizado por Reforço, Multi-armed Bandit; Agrupamento; K-means.

List of Abbreviations and Acronyms

AH – Alternate Heuristic

AM – Additive Manufacturing

A-MAT – Adaptive Medial Axis Transformation

AMF – Additive Manufacturing File

ASCII – American Standard Code for Information Interchange

AWS – American Welding Society

AI – Artificial intelligence

BH – Biased heuristic

CAD – Computer -Aided Design

CC – Convergence Criterion

CH – Continuous Heuristic

CMT – Cold Metal Transfer

CTSP – Clustering Travelling Salesman Problem

CTWD – Contact Tip to Work Distance

DED-MAM – Direct Energy Deposition Metal Additive Manufacturing

EB-DED-MAM – Electron Beam Direct Energy Deposition Metal Additive Manufacturing

FDM – Fused Deposition Modelling

GDP – Gross Domestic Product

GMAW – Gas metal arc welding

GRASP – Greedy Randomized Adaptive Search Procedure

GTAW – Gas tungsten arc welding

GTSP – Generalization Travelling Salesman Problem

H – Hypothesis

HTP – Heuristic of Trajectory Planning

IC – a trajectory generation node started at a position on the contour

L-DED-MAM – Laser Direct Energy Deposition Metal Additive Manufacturing

MAB – Multi-Armed Bandit

MAM – Metal Additive Manufacturing

MAT – Medial Axis Transformation

MJM – Multi-Jet Modelling

NIAC – Near-Immersion Active Cooling

NNH – Nearest Neighbor Heuristic

AO – Axis Ordering

OC – a trajectory generation node started outside (inward) the contour

PAW – Plasma arc welding

PB-MAM – Powder Bed Metal Additive Manufacturing

PPPS – Printing Process Planning Software

RCH – Random Contour heuristic

RL – Reinforcement Learning

SAE – Society of Automotive Engineers

SL – stereolithography

SLS – Selective Laser Sintering

SLM – Selective Laser Melting

SQ – Scientific Question

SO – Specific Objective

SSE – Sum of Squared Errors

TSP – Traveling Salesman Problem

TSPLIB – Traveling Salesman Problem Library

TS – Thompson Sampling

UCB – Upper Confidence Bound

WAAM – Wire arc additive manufacturing

Table of Contents

CHAPTER I

INTRODUCTION

1.1 Contextualisation

Efficient and sustainable manufacturing has become a priority for companies to ensure profitability and compliance with environmental regulations. One such alternative to meet these requirements is additive manufacturing (AM), also known as 3D printing. Additive manufacturing is defined by ISO/ASTM 52900:2018 as “a process of joining materials to make parts from 3D model data, usually layer by layer, as opposed to subtractive manufacturing and formative manufacturing methodologies”. This manufacturing approach, according to Aljarrah et al. (2020), is presented under several technologies, depending on the type of material used (plastic, metal, or concrete). Examples of such technologies are fused deposition modelling (FDM), stereolithography (SL), selective laser sintering (SLS), selective laser melting (SLM), multi-jet modelling (MJM), among others.

This industrial manufacturing process has the potential to significantly reduce resource and energy demands, as well as CO_2 emissions per unit of Gross Domestic Product (GDP), according to Gebler et al. (2014). When compared with subtractive methodologies, Yuan et al. (2020) highlighted advantages such as low cost, reduction in material usage, and productivity improvement. Due to these advantages, Matos et al. (2020) pointed out that AM has been used in various areas, such as medical sciences (medical and dental implants), jewellery, tennis, automotive, and aerospace industries. According to Ding et al. (2015a), AM becomes more competitive when it comes to expensive materials like titanium and nickel alloys, which are commonly used in the aerospace industry. This is because it can significantly reduce the buy-to-fly ratio, also known as the buy-to-apply ratio, when applied in other fields.

In the manufacturing of metallic components, AM can also achieve satisfactory results in terms of sustainability and efficiency, according to Wippermann et al. (2020), Jackson et al. (2016), and Bekker and Verlinden (2017). Several techniques are available for AM of metal parts, each based on dedicated layer building methods, heat sources, and feeding forms. In an attempt to rationalise the technique description, Paolini et al. (2019) cited that metal additive manufacturing (MAM) has been ordered as by powder bed (PB-MAM) and by direct energy deposition (DED-

MAM). Also, according to the authors, in these technologies, a heat source (laser, arc, or electron beam) fuses a feedstock in the form of powder or wire. Colomo et al. (2020) stated that PB-MAM is required when targeting complex shapes printed parts, with dimensional accuracy and good surface quality. PB-MAM technologies achieve the complement of this. Cooke et al. (2020) highlighted that DED-MAM has gained the attention of industries and universities worldwide for manufacturing medium to large parts and with good structural integrity, achieving high printing speeds and still being suitable for repairs.

Further detailing the DED-MAM techniques, the parts are printed employing an energy source that melts the raw material (wire or powder) while it is being deposited. The most common energy sources used for that are laser (L-DED-MAM), electron beam (EB-DED-MAM), welding arc (usually the same used in plasma welding, gas tungsten arc welding, and gas metal arc welding). Chen et al. (2020) reported that the EB-DED-MAM technique, carried out in a vacuum camera, is ideal for printing reactive alloys; this process can become expensive for other applications. However, current literature and industrial applications suggest EB-DED-MAM beyond reactive alloys. According to the comparison presented by Negi et al. (2019), L-DED-MAM is an alternative recommended for printing parts accurately with high dimensional tolerances. They also pointed out L-DED-MAM disadvantages, such as high investment and operational costs and low deposition rates. According to Ding et al. (2015a), both processes cited do not have remarkable energy efficiency; EB-DED-MAM processes with values of 15 % to 20 % and L-DED-MAM with values of 2 % to 5 %, which are low compared to arc welding, which can be more efficient than 90 %.

Using arc welding as the energy source, the DED-MAM techniques are referred to as Wire arc additive manufacturing (WAAM). Gas metal arc welding (GMAW), Gas tungsten arc welding (GTAW), and Plasma arc welding (PAW) are the most common welding processes from which the energy source is used in DED-MAM. In short, WAAM consists of an electric arc as a heat source for melting a wire, enabling continuous deposition of material, layer by layer, until the end of the printed part. According to Jafari et al. (2021), one of the benefits of using GMAW-based WAAM (as we advance, the term WAAM will be used to refer to GMAW-based WAAM) in all is the ability to produce large components or parts (ranging from 1000 mm to 3000 mm) with a reasonable level of geometric complexity and dimensional accuracy. In complement to the benefits, Cunningham et al. (2018) mentioned that WAAM has high deposition rates and allows working with various metals. The major point is that WAAM is based on well-established welding technology, enabling the application of several already-known techniques.

The promise of printing medium to large-sized parts, achieving energy savings, high deposition rates, and reduced material usage has led to an increasing number of companies adopting WAAM technology and rising of investments in research and development to explore WAAM potential further. Regarding the companies, MX3D's M1 Metal AM System and Gefertec's arc60x and arc40x hardware, paired with 3DMP-CAM software, are examples of complete solutions that enable the printing of medium to large-sized parts. MX3D and Gefertec are responsible for printing the parts depicted in Figure 1.1(a) and 1.1(b), respectively. RAMLAB is a renowned laboratory for additive manufacturing in the offshore and maritime sectors, and has already successfully printed large-sized parts, including a ship propeller with a weight of 400 kg and diameter of 1350 mm, as depicted in Figure 1.1(c). Regarding the investments, according to article available on the Metal AM website (2021), the Innovate UK has provided financial support of £1.2 million for an academic-industrial project aimed at leveraging high productivity with WAAM, with a focus on adopting the technology in industries such as construction and resources. This is not an isolated example; the second article on the 3D printing industry website (2020), reported that the oil and gas industry consortium has also invested in the production and qualification of parts for the sector.

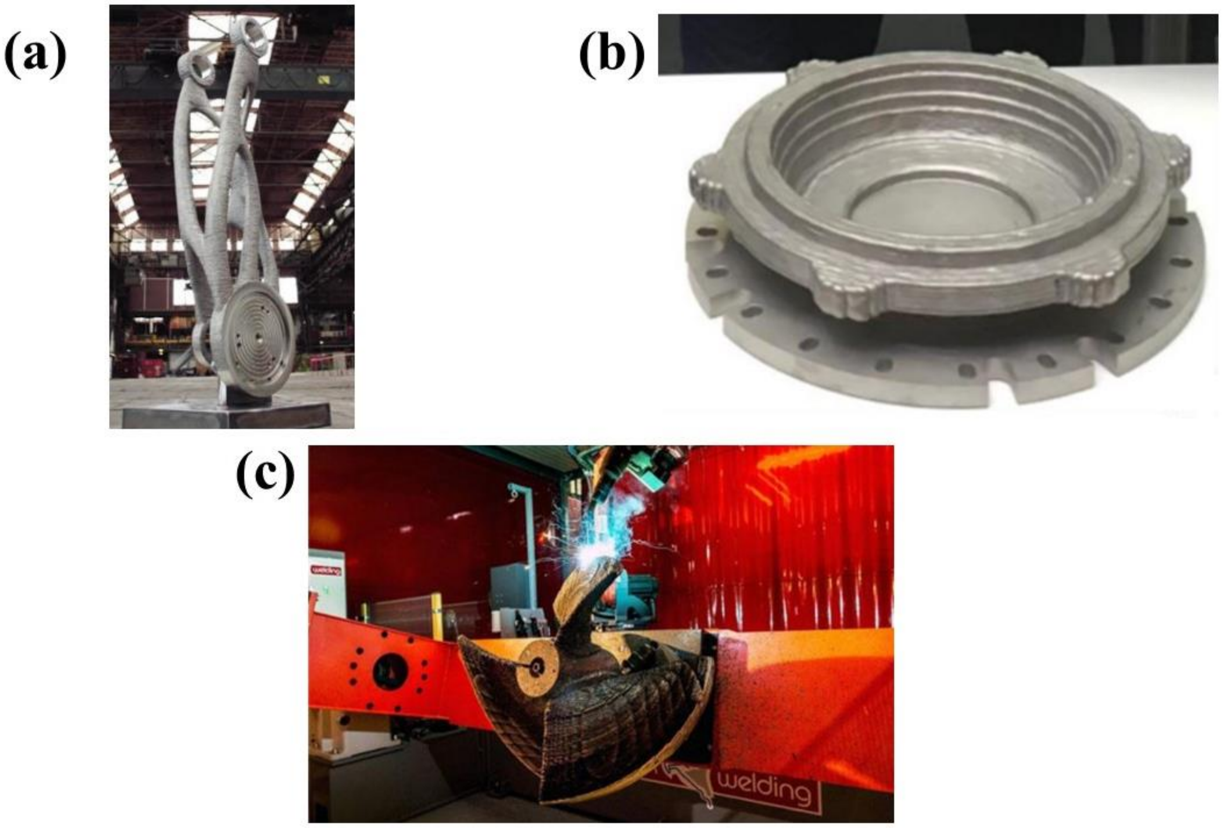


Figure 1.1 - Printed parts by WAAM: (a) robot arm with optimized topology (METAL AM, 2020); (b) wheelset bearing cover (METAL AM, 2019); and (c) ship propeller (RAMLAB, 2017)

However, despite the examples presented of the use of WAAM for various sectors and its potential for manufacturing metallic components, this technique still has certain limitations that restrict its acceptance by the industry. According to Singh et al. (2021), a crucial step for this acceptability to occur is good process planning, which is an important task for implementing and properly using any manufacturing process.

Among all the process planning steps in WAAM, the trajectory planning (also known as path planning) for the deposition torch is one of the key factors for the success of this technology, although not always commented on, and sometimes even neglected. Trajectory planning, being a key factor, in turn, encompasses some complexity. According to Rodrigues et al. (2019), mediocre planning may result in porosity, internal defects, lack of fusion between adjacent beads and high residual stresses. Debroy et al. (2018) cited that printed part anisotropy can be avoided and/or eliminated with good strategies for torch path. Some traditional trajectory planning, commonly used in polymer additive manufacturing, depends on the part complexity and is inappropriate for WAAM, as seen in Figure 1.2.

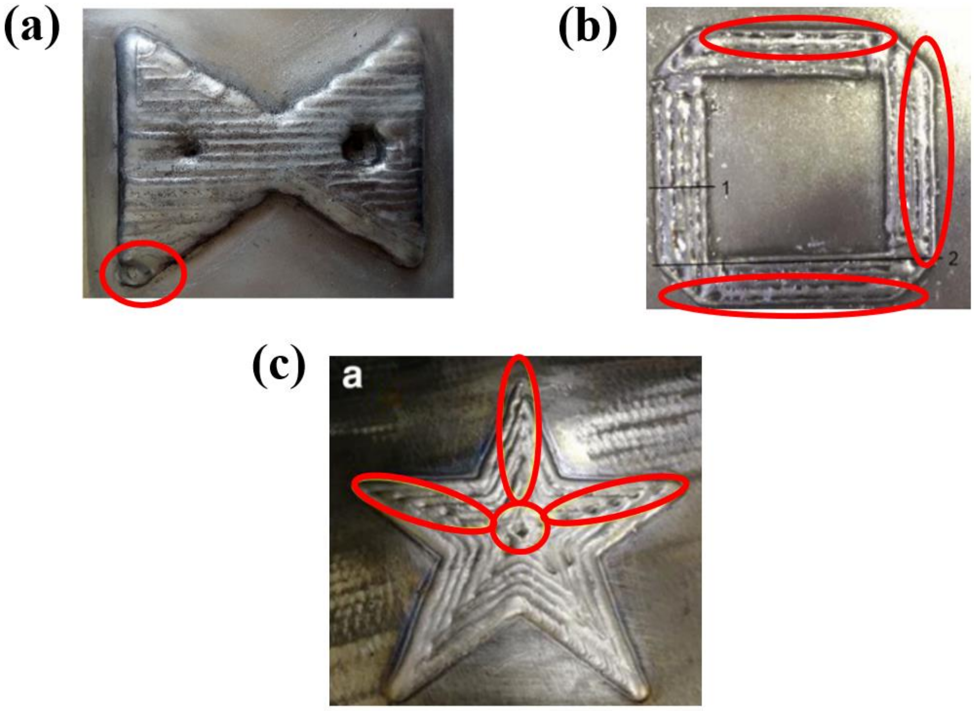


Figure 1.2 - Limitations of conventional WAAM strategies leading to non-conformities: (a) material shortage resulting from Zigzag strategy (Wang et al., 2010); (b) unfilled area resulting from Hydrid (Contour+Zigzag) strategy (Ding et al., 2014); and (c) unfilled area resulting from Parallel Contour strategy (Liu et al., 2020)

Highlighting the challenge of material shortage in the arc extinguishing area during WAAM, Figure 1.2(a) shows the issue in a part printed using the Zigzag strategy. This well-documented problem arises due to the lack of a closed-loop path (the starting point differs from the ending point). With this approach, this shortcoming can be mitigated, material accumulation at the starting point counteracts the shortage at the ending point. Moreover, the repetition of the arc extinguishing point in multiple layers of the printed part exacerbates this geometric non-conformity. In Figure 1.2(b), unfilled areas (which are occasionally referred to as voids) are shown in a part printed using the Hybrid strategy (Parallel Contour + Zigzag strategy). This problem can occur when the part dimensions are not multiples of the distance between the beads configured to generate the trajectory, which is a common situation in more complex parts. As a result, a missed scan line in the trajectory can become an unfilled area in the printed part. In Figure 1.2(c), unfilled regions are visible in a printed part using the Parallel Contour strategy. Liu et al. (2020) explain this is due to a trajectory with a direction change (usually with an angle smaller than 58.65°), which generates a non-constant offset trajectory, particularly at the direction change point. As a result, adjacent beads deposited along this trajectory cannot overlap their neighbours, resulting in an unfilled area. Chapter II will provide more details on this issue.

Besides unfilled regions, conventional strategies can often lead to a high number of non-deposition movements, which can significantly increase build time, potential arc re-ignition issues and non-conformities in the deposited bead geometry. This is highlighted in Figure 1.2(a), where the deposited bead exhibits significant deviations from the intended shape. These issues prompt the researchers to explore novel strategies that can overcome these limitations and deliver superior-quality parts more efficiently. To address these issues, the use of Space-Filling strategies has emerged as a promising solution. While this approach has shown success in other additive manufacturing technologies, its potential in the context of WAAM has yet to be fully explored.

Cox et al. (1994) define Space-Filling strategies (or curves) as continuous trajectories in a unit square that passes through all the points that discretise this square. This type of strategy solves the filling problem because, as Sebastian et al. (2020) explained, fractals (mathematical space-filling curves) can well define a path to fill a given surface. Regarding the high number of non-deposition movements, Kapil et al. (2016) developed three Space-filling strategies for additive manufacturing that can reduce and even zero the amount of non-deposition movements to an arbitrary area. However, some of these strategies generate material accumulations, that is, an imperfection that subtractive processes must remove after building each layer. For polymer additive manufacturing, Lin et al. (2019) also developed a Space-filling strategy called maze-like, which generates trajectories with non-deposition movements reduced for a more complex area. In addition to mitigating filling problems and many non-deposition movements, Catchpole-Smith et al. (2017) explain that Space-Filling strategies can potentially provide a more uniform temperature distribution than the cyclic heat input associated with one-way straight lines (characteristics of conventional strategies). This finding was updated to WAAM in studies conducted by Vishwanath and Suryakumar (2022), where it was evidenced that fractals (another Space-Filling strategy) generate higher temperature uniformity and, consequently, lower distortion in the built part when compared with conventional strategies. Recently, Singh et al. (2022) achieved good results in print quality using Space-Filling strategies in WAAM using a Travelling Salesman Problem (TSP) solver. By strategically positioning grid points, the proposed path planning method was able to orient the toolpath in a way that favoured one direction of motion over the other. By implementing Space-Filling strategies, it is possible to reduce the number of non-deposition movements and minimize the occurrence of non-conformities, resulting in a more efficient and reliable printing process.

While Space-Filling strategies have been proven successful in other AM technologies, their implementation in WAAM has been met with mixed results. Some related works have cited negative approaches, such as increased printing time and poor quality (Michel et al., 2019, and Sun et al., 2023) due to constant direction changes, and high processing times for generating the trajectory (Ding et al., 2015b). On the other hand, the positive results achieved in WAAM have primarily been demonstrated on simple parts such as cubes and animal shapes, so further evidence is required to fully validate the efficacy of this strategy across a broader range of complex geometries. Due to these factors, the literature on the appropriateness of Space-Filling strategies in WAAM remains unclear. In light of these challenges, the following research question arises: is it possible to effectively implement a Space-Filling strategy in WAAM?

1.2 Global Objective

Building upon the open question raised earlier, the primary objective of this thesis is to systematically investigate the potential benefits and challenges of implementing a Space-Filling strategy in WAAM. This will be achieved by creating and refining an innovative strategy, as well as conducting extensive computational and experimental testing to evaluate its performance (concerning geometrical features and printing operationality). Through this study, it is expected to provide valuable insights into the advancements and possibilities of WAAM technology and lay the foundation for the wider adoption of Space-Filling strategies in the industry.

It is important to state that this proposal does not intend to tackle the metallurgical and mechanical property aspects of printing parts using WAAM. The target is to optimise the printing features (trajectory planning, not production planning). However, the proposal outcomes are, by principle, applicable to any metallic material, as long as the right parametrisation and property qualification are carried out in advance and/or in parallel. Like welding operations, production planning involves the process and the metallurgical aspects, and one must adapt to the other.

One limitation of the available trajectory planners, including the one proposed in the global objective of this work, is that the parameter selection is carried out separately. However, the proper parameters are needed to print the part in the designed dimension and avoid material concentration in crossings, faults between layers, inadequate start-stops, etc. The software users of trajectory planners must be aware of it and have close contact with the operational personnel (not always working together).

Figure 1.3 illustrates this principle where parametrisation (Module “Parametric design and parameter definition”) works synergically with trajectory planning (Module “Trajectory planning and machine code generation”). The end-user needs to enter the parameters to build the part of the component with the desired dimension of each track. Optimisation routines can facilitate finding the optimal parameters for a given track width, according to the end user’s restriction definition (acceptable tolerance, hardness, etc). Using machine learning concepts will permit feedback to the database, increasing the customised process’s robustness. But the development of this complete concept (production planning) is a subject beyond the scope of this thesis.

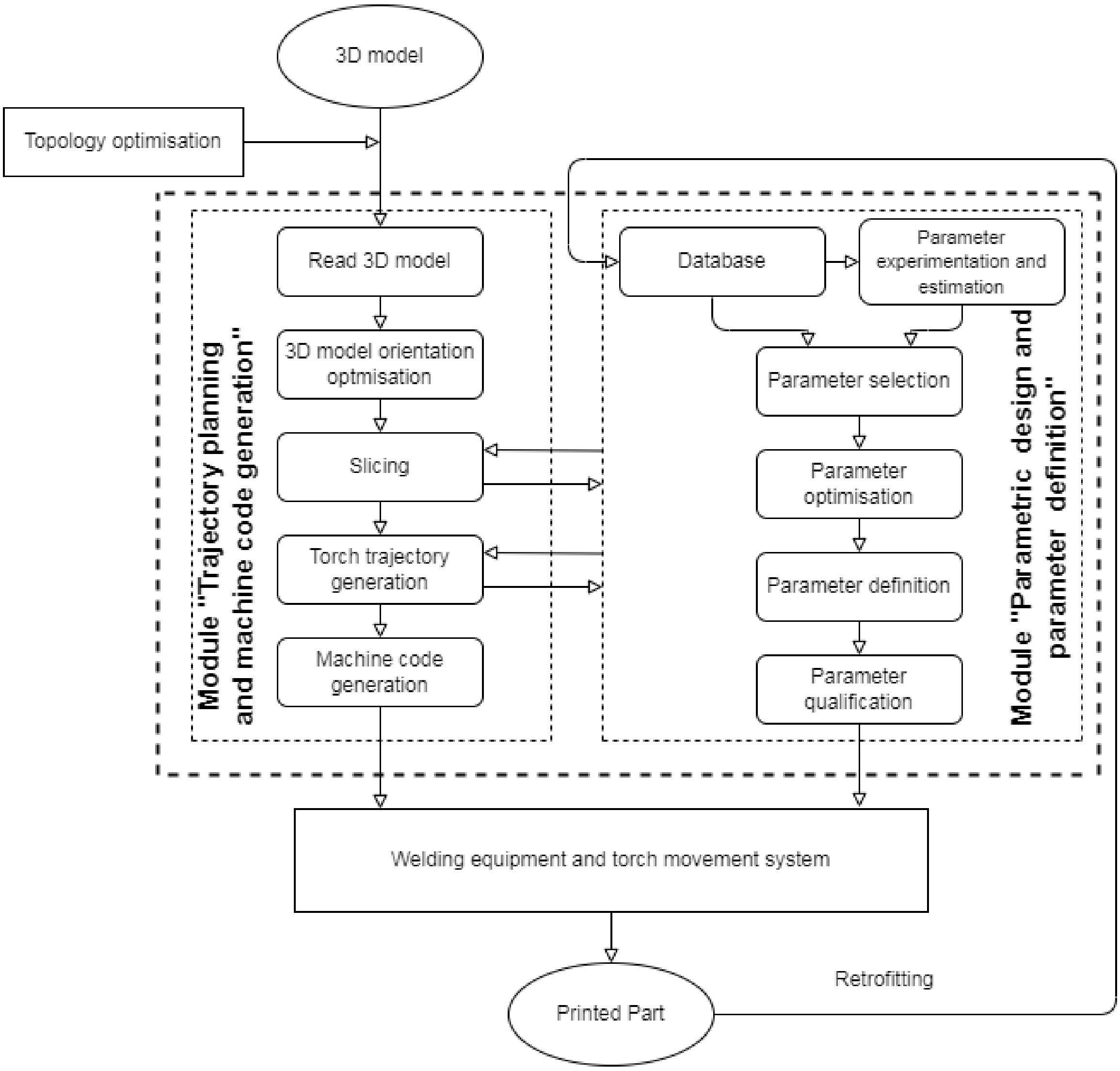

3D model
Topology optimisation
Read 3D model
3D model orientation optmisation
Slicing
Torch trajectory generation
Machine code generation
Module "Trajectory planning and machine code generation"
Database
Parameter experimentation and estimation
Parameter selection
Parameter optimisation
Parameter definition
Parameter qualification
Module "Parametric design and parameter definition"
Welding equipment and torch movement system
Printed Part
Retrofitting

Figure 1.3 – The integration of trajectory planning and generation with parametrisation to compose a production plan for WAAM

1.3 Structure of the thesis

This thesis can be thought of as an additive manufacturing process, where a complete solution for the utilization of a Space-Filling strategy in WAAM is built layer by layer. It is composed of eight chapters, with each chapter representing a layer in the overall solution.

In the substrate or print table, we have the Introduction (Chapter 1), which provides a contextualisation, objective, and structure of the thesis as a foundation for the rest of the layers. In subsequent layers (Chapters 2 to 5), a novel Space-Filling strategy is developed and improved upon step-by-step, with each layer building upon the previous one, as follows:

- In the first layer (Chapter 2), the concept of the novel Space-Filling strategy for WAAM is introduced, and its algorithm is presented and validated through experimental testing. The potential advantages of the proposed strategy over conventional Space-Filling strategies are also highlighted.
- In the second layer (Chapter 3), an improved algorithm is developed to address the problems left unresolved in the first implementation. The algorithm is validated and compared with conventional strategies to provide a more robust and effective solution.
- In the third layer (Chapter 4), a reinforcement learning algorithm is used to improve the computational time and response quality of the previous strategy introduced in Chapter 3. The algorithm is developed and validated in comparison with the previous strategy, and a case study is performed to demonstrate the effectiveness of the improvements.
- In the fourth layer (Chapter 5), a clustering algorithm is used to further enhance the computational time and response quality of the previous strategy introduced in Chapter 4. A validation test is performed to demonstrate the effectiveness of the complete solution. This layer aims to demystify the idea of a long process time to generate a Space-Filling strategy.
- In the fifth layer (Chapter 6), general conclusions are presented, divided according to the specific objectives presented in Chapters 2 to 5 but presented in an interrelated way with the general objective of the work.

Complementing the layout of this manuscript, in the sixth layer (Chapter 7), suggestions for future work are included. Finally, in the seventh layer (Chapter 8), bibliographical references are presented.

Therefore, by layering upon each other, our proposal is that this thesis can present a comprehensive and effective approach towards utilizing Space-Filling strategies in WAAM.

In addition to this thesis, several scientific articles were produced during the doctoral period. These articles are based on the computational and experimental work carried out in the context of the present thesis. Moreover, there is a related list of studies that were conducted in parallel, and which share a common theme with the present work.

Articles published in congresses, seminars, and conferences:

a) FERREIRA, RAFAEL PEREIRA; SCOTTI, AMÉRICO. Planejamento de trajetórias para manufatura aditiva por deposição a arco implementada em software aberto. Anais do Congresso Internacional de Engenharia Mecânica e Industrial. Anais. Brasília (DF), 2020.

DOI: https://10.29327/conemi.290013

b) FERREIRA, RAFAEL PEREIRA; SCOTTI, AMÉRICO. Fluxo de trabalho da impressão 3D por MADA. Anais do I Congresso Brasileiro de Manufatura Aditiva, 2020

c) FARIAS, RODRIGO MARTINS; FERREIRA, RAFAEL PEREIRA; VILARINHO, LOURIEL OLIVEIRA. Desenvolvimento de um gerador automático de modelos a partir do código G para simulações numéricas do processo de manufatura aditiva por deposição a arco. 11th Brazilian Congress on Manufacturing Engineering, 2021. DOI: https://10.26678/ABCM.COBEF2021.COB21-0084

d) FERREIRA, RAFAEL PEREIRA; SCOTTI, AMÉRICO. Otimização do Planejamento de Trajetórias na Manufatura Aditiva por Deposição a Arco – Uma Análise Estatística da Estratégia Pixel. In: Proceedings of the 2nd (ICAIC) International Conference for Academia and Industry Co-operation & 2nd (IMMSEM) International Meeting in Materials Science and Engineering of Maranhão. Anais.São Luís(MA) IFMA, 2021. DOI: https://10.29327/2ndicaic2ndimmsem2020.438487

Articles in peer-reviewed and indexed journals:

1. Published:

a) FERREIRA, R.P.; SCOTTI, A. The Concept of a Novel Path Planning Strategy for Wire + Arc Additive Manufacturing of Bulky Parts: Pixel. **Metals** 2021, 11, 498. https://doi.org/10.3390/met11030498

b) FERREIRA, R.P., VILARINHO, L.O., SCOTTI, A. Development and implementation of a software for wire arc additive manufacturing preprocessing planning: trajectory planning and machine code generation. **Weld World** 66, 455–470, 2022. https://doi.org/10.1007/s40194-021-01233-w

2. Submitted:

a) FERREIRA, R.P., VILARINHO, L.O., SCOTTI, A. Enhanced-Pixel Strategy for Wire Arc Additive Manufacturing Trajectory Planning: operational efficiency and effectiveness analyses. This work has been submitted to the Rapid Prototyping Journal and is currently awaiting a final decision on publication.

3. To be Submitted to a journal:

a) FERREIRA, R.P., SHUBERT, E., SCOTTI, A. An Advanced Pixel Path Planning Strategy for Wire Arc Additive Manufacturing applying a reinforcement learning approach.

b) FERREIRA, R.P., SHUBERT, E., SCOTTI, A. Fast Advanced-Pixel: a k-means clustering path planning solution for Wire Arc Additive Manufacturing.

CHAPTER II

THE CONCEPT OF A NOVEL PATH PLANNING STRATEGY FOR WIRE+ARC ADDITIVE MANUFACTURING OF BULK PARTS: PIXEL

2.1 Introduction, Scientific Questions, and Specific Objectives

Wire arc additive manufacturing offers the ability to produce components with varying wall thicknesses, from thin to thick. However, there is still no consensus on optimising production efficiency and effectiveness, and even the terminology used to describe wall thickness can be unclear. For the purposes of this work, thin walls are defined as those produced using a single track per layer, with or without torch oscillation. In contrast, thick walls require multiple tracks per layer or a single track with wide-amplitude oscillation (known as Zigzag). Confusion can arise when using torch oscillation in a single track, such as with a rectangular profile, and the Zigzag strategy. The author of this work defines oscillation as occurring when the melting pool remains liquid for the entire oscillation amplitude, while the Zigzag strategy involves solidification during the transversal movement of the arc. Therefore, while the Zigzag strategy may resemble a single track per layer, it is not strictly defined as such due to the occurrence of solidification during the movement.

To manufacture thick-wall parts using WAAM, it is essential to consider that the deposition strategies used for polymer building may not be effective due to the different nature of the trail width obtained by these technologies. In the case of WAAM, wider spaces between trails are necessary due to the width of the beads achieved by various welding processes and techniques. For example, Cui et al. (2021) deposited a nickel-aluminium-bronze alloy and obtained layers (with a single pass, or track, deposition per layer) with dimensions ranging from 3.2 to 9.0 mm in width and from 2.6 to 4.5 mm in height using the GMAW-CMT and GMAW-pulsed processes, respectively. Teixeira et al. (2022) also developed a working envelope with a single deposition pass per layer using GMAW-CMT to cover layer dimensions ranging from 4.8 to 8.1 mm in width and 1.7 to 3.5 mm in height with carbon steel. Larger layer widths can be achieved with oscillation techniques, as shown in the study by Ma et al. (2019) or by the use of double wire, as presented in the studies by Martina et al. (2019) and Shi et al. (2019). Conversely, polymers can achieve smaller widths than 1 mm. However, with larger layer widths, conventional building strategies may result in non-conformities, such as unfilled regions, which are often not noticeable in the building of polymers due to their narrower track width.

To further elaborate on the issue, Figure 2.1 depicts two deposition simulations using the Parallel Contour strategy with wide trails commonly obtained by WAAM, and the Zigzag strategy as a thick wall-filling strategy. As shown in the zoomed-in details in the drawings, both strategies can present filling problems. In the Parallel Contour strategy, unfilled regions can be observed (left frame). Liu et al. (2020) explain this is due to a trajectory (white arrow) with a direction change (usually with an angle smaller than 58.65°), which generates a non-constant offset trajectory (yellow arrow), particularly at the direction change point. As a result, adjacent beads deposited along this trajectory cannot overlap their neighbours, resulting in an unfilled area. This issue is exacerbated as the width of the beads increases. Additionally, torch movements with non-deposition (represented by the red dashed lines) can increase building times and cause non-conformities in the bead geometry due to arc strikes and stops, as demonstrated by Hu et al. (2018).

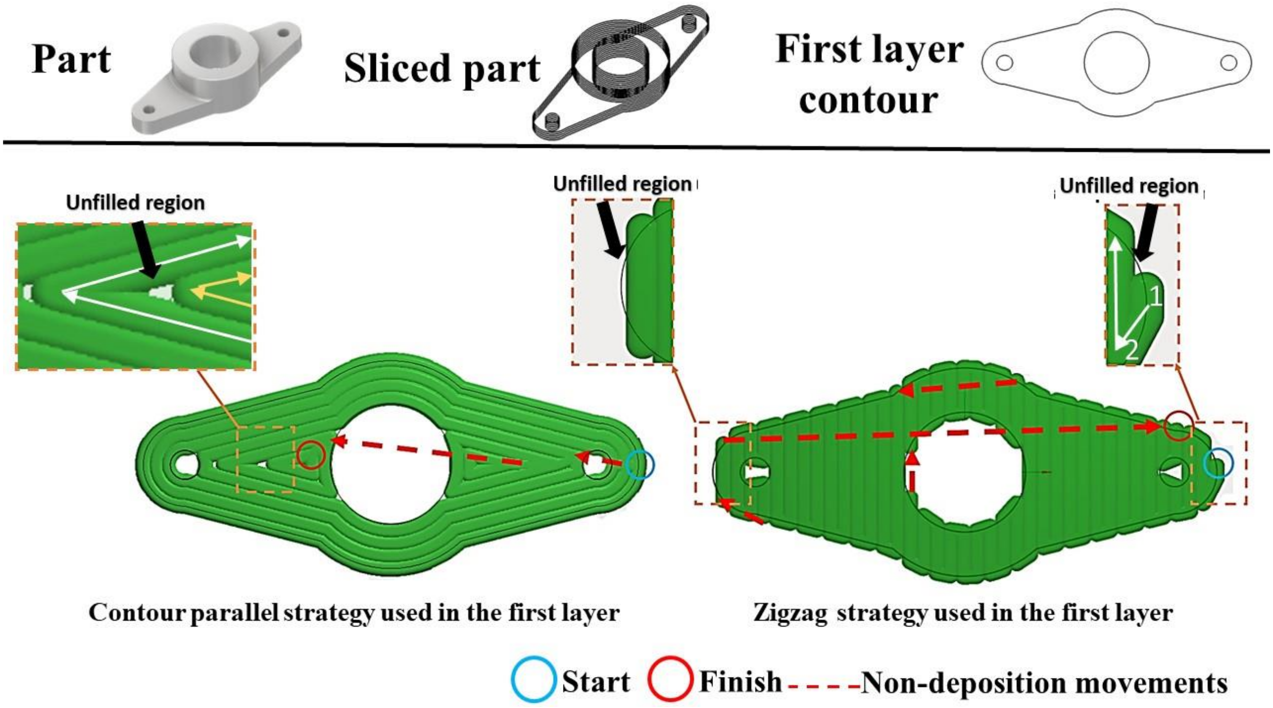

Part
Sliced part
First layer contour
Unfilled region
Unfilled region
Unfilled region
1
2
Contour parallel strategy used in the first layer
Zigzag strategy used in the first layer
Start
Finish
- - - -Non-deposition movements

Figure 2.1 - Building simulations of a part with Parallel Contour (left) and Zigzag (right) strategies, focusing on non-conformities

Similarly, the Zigzag strategy can also present filling problems. The first issue occurs when the part's dimensions are not a multiple of the distance between the configured beads to generate the trajectory, a common situation in complex parts, resulting in unfilled regions, as illustrated in the central frame of Figure 2.1. The second issue arises with circular features in the part, where the trajectory always starts at the most external points (represented by point 1 in the right frame), and its direction changes asides (represented by point 2 in the right frame), leading to filling problems on the opposite side. Non-deposition movements can also occur in this strategy. These conventional strategies can lead to unfilled regions and a high number of non-deposition movements, resulting in an increase in building time and non-conformities in the deposited bead geometry. Furthermore, the complexity of the part can aggravate the filling problems mentioned above.

As mentioned in the introduction, the Space-filling strategy has shown promising as a solution to address these issues. However, it is important to note that the previously mentioned strategies, along with others that may be applicable but not yet widely reported, were developed as solutions for specific cases of deposition flaws in layer building. Replicating these solutions in other geometries can pose new setbacks, and there is room for optimizing trajectory planning by considering factors such as trajectory distance and the number of stops and restarts. Many existing strategies are based on a single solution with low path flexibility, with adjustments typically limited to trial and error, such as swapping from inward to outward directions in the contour strategy or changing the angle in the Zigzag strategy. Therefore, there is ample opportunity to develop new trajectory generation strategies tailored explicitly for WAAM.

Given the above, a first scientific question (SQ) arises with its respective hypothesis (H):

- SQ 2.1 - Can a novel Space-Filling strategy for WAAM be developed using trajectory optimization algorithms to provide greater flexibility and applicability to more complex geometries beyond a single-case solution?
- H 2.1 - By developing a new Space-Filling strategy for WAAM with the aid of trajectory optimization algorithms, it is possible to increase the flexibility of the strategy and enable its applicability in a wider range of complex geometries beyond a single case solution. This will reduce deposition flaws and non-conformities, leading to improved quality and efficiency in WAAM processes.

Considering the scientific question and hypothesis mentioned, the following specific objective (SO) were proposed:

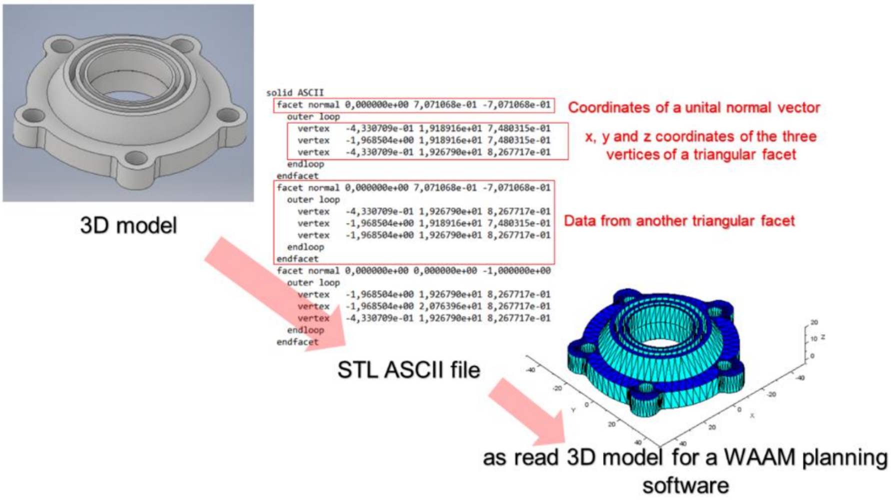
- SO 2.1 - The objective is to develop a new Space-Filling strategy for WAAM that offers greater flexibility in trajectory planning for complex geometries using trajectory optimization algorithms, and to evaluate its effectiveness.

The concept of flexibility in this context refers to providing multiple solutions for printing a single layer and allowing the user to choose the path width without worrying about deposition issues, unlike conventional strategies. This will be further clarified during the presentation of the results. The concept of applicability is the effectiveness of the new Space-Filling strategy in the context of WAAM technology.

2.2 Literature Review: Path planning strategies in WAAM

Trajectory planning or path planning (these two terms might be used interchangeably throughout this work) is a critical step in the process of printing a part. All the process involves creating a 3D model preprocessing it, slicing it into layers, and generate the trajectory (the path planning step) for each layer and then generating the machine code to print a part. Part printing involves using a 3D model created in CAD software or obtained through a 3D scan. The model can be manipulated and exported in common file formats such as STL or AMF (it's worth noting that there are other file formats available for additive manufacturing) with STL being the most commonly used format. STL files can be encoded in either binary or ASCII, but binary files are preferred due to their smaller file size. The STL format consists of triangular facets with x, y, and z coordinates of vertices and a normal vector to indicate the external facet side, which is essential for the accurate reproduction of the model. Figure 2.2 illustrates the workflow that transforms a 3D CAD model into a processable input format for the subsequent steps.

During the slicing process in WAAM, the orientation of the part is crucial to ensure that each layer is adequately supported by the previous one. Unlike polymer additive manufacturing, it is not possible to use supports in WAAM. Therefore, the user must carefully position the part to ensure that the printing process is stable and accurate. This is demonstrated in Figure 2.3, which illustrates the importance of proper orientation in achieving successful prints. During the slicing process, the most common approach involves creating cutting planes parallel to each other and perpendicular to the slicing axes, which intersect the 3D model. Typically, equidistant slicing planes are used, although non-equidistant planes are also possible. The intersection coordinates are determined using the known coordinates of the vertices of the triangular mesh in the STL file. The two-dimensional contours of each layer are reached by connecting the endpoints to starting points of a line between two interceptions (due to adjacent triangles sharing the same side, the endpoint of an interception line is always coincident with the starting point of a subsequent interception line). As standardised, the outer contours were generated with the counterclockwise vertex disposition, while the internal contours were arranged clockwise. For more details about this process, please see Ferreira et al. (2022).



```
solid ASCII
  facet normal 0,000000e+00 7,071068e-01 -7,071068e-01
    outer loop
      vertex   -4,330709e-01 1,918916e+01 7,480315e-01
      vertex   -1,968504e+00 1,918916e+01 7,480315e-01
      vertex   -4,330709e-01 1,926790e+01 8,267717e-01
    endloop
  endfacet
  facet normal 0,000000e+00 7,071068e-01 -7,071068e-01
    outer loop
      vertex   -4,330709e-01 1,926790e+01 8,267717e-01
      vertex   -1,968504e+00 1,918916e+01 7,480315e-01
      vertex   -1,968504e+00 1,926790e+01 8,267717e-01
    endloop
  endfacet
  facet normal 0,000000e+00 0,000000e+00 -1,000000e+00
    outer loop
      vertex   -1,968504e+00 1,926790e+01 8,267717e-01
      vertex   -1,968504e+00 2,076396e+01 8,267717e-01
      vertex   -4,330709e-01 1,926790e+01 8,267717e-01
    endloop
  endfacet
```

3D model
Coordinates of a unital normal vector
x, y and z coordinates of the three vertices of a triangular facet
Data from another triangular facet
STL ASCII file
as read 3D model for a WAAM planning software

Figure 2.2 - Model transformation steps from a 3D model CAD drawing to the processable input format for printing

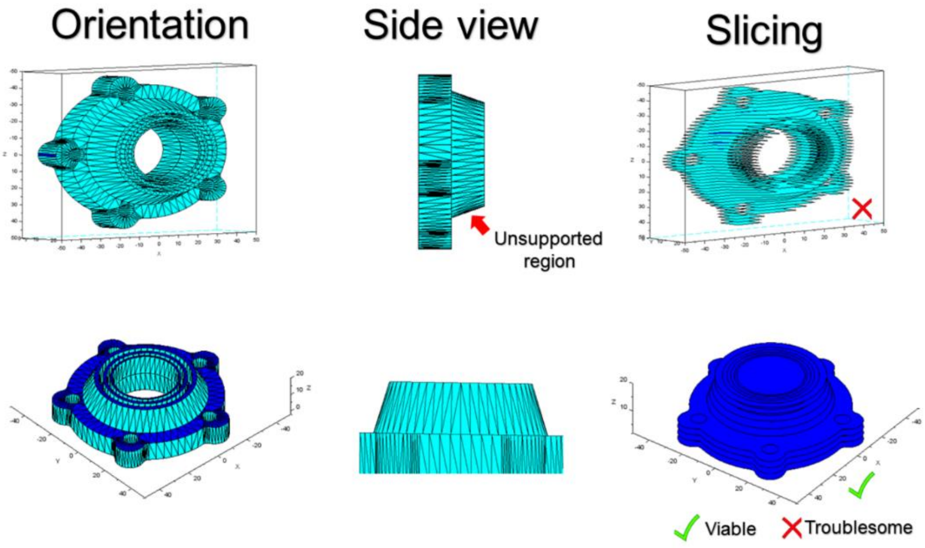
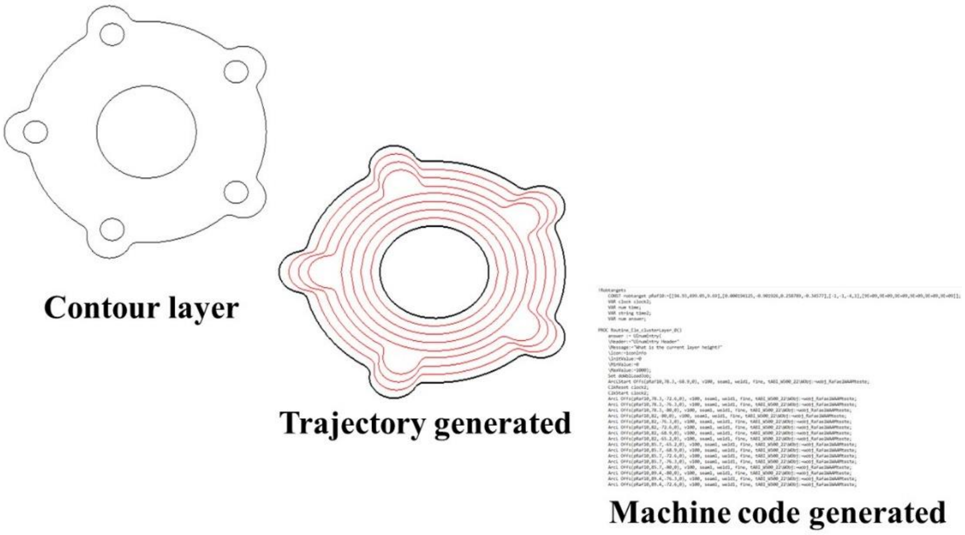


Figure 2.3 - Potential orientations for a same three-dimensional model (Fig. 2.2) and the consequences on the slices of the model: upper view = troublesome; lower view = viable

The next step in the workflow involves planning the trajectory for each slice. This is done using a set of vectors that determine the torch movement during printing. The resulting coordinates are organized into a 3-column matrix (for the case to print using only 3 axes). The information from this step is then used to generate machine code (a G-code or a robot code), with the necessary commands automatically inserted into the code header. An example of the trajectory and machine code generated can be seen in Figure 2.4.



```
!Robtargets
    CONST robtarget pRaf10:=[[94.93,499.05,9.69],[0.000194125,-0.901926,0.258789,-0.34577],[-1,-1,-4,1],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]];
    VAR clock clock2;
    VAR num time;
    VAR string time2;
    VAR num answer;

PROC Routine_Ele_clusterLayer_0()
    answer := UInumEntry(
    \Header:="UInumEntry Header"
    \Message:="What is the current layer height?"
    \Icon:=iconInfo
    \InitValue:=0
    \MinValue:=0
    \MaxValue:=1000);
    Set doWblLoadJob;
    ArcLStart Offs(pRaf10,78.3,-68.9,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ClkReset clock2;
    ClkStart clock2;
    ArcL Offs(pRaf10,78.3,-72.6,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,78.3,-76.3,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,78.3,-80,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,82,-80,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,82,-76.3,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,82,-72.6,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,82,-68.9,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,82,-65.2,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,85.7,-65.2,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,85.7,-68.9,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,85.7,-72.6,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,85.7,-76.3,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,85.7,-80,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,89.4,-80,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,89.4,-76.3,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf10,89.4,-72.6,0), v100, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
```

Figure 2.4 – The trajectory (represented in red) for the first contour layer of the 3D model (introduced in Figure 2.2), along with the corresponding machine code

To plan the path for printing a part, in short, the cross-section of the layers delineates geometric shapes with different sizes and outline complexities. In relation to bulky parts, geometric complexity is an obstacle to be overcome. An example of printing difficulties is in slices composed of nonconvex cross-sectional polygons, with or without obstacles (such as through or blind holes of different shapes). According to Gerdjikov and Wolff (2008), convex polygons are those in which every internal angle is less than 180°, as illustrated in Figure 2.5(a). Note that in the nonconvex configuration, Figure 2.5(b), at least one internal angle is between 180° and 360°, so that points of a line segment between two points on the polygon boundary of the polygon can be located outside the polygon. In the case of nonconvex geometries, inefficient trajectories can lead to voids inside the printed part (LIU et al., 2020). When holes are presented, “empty” trajectories (paths with no deposition) are used to avoid this obstacle; low surface quality due to geometric irregularities caused by frequent arc extinction and reignition are the consequences (HU et al., 2018).

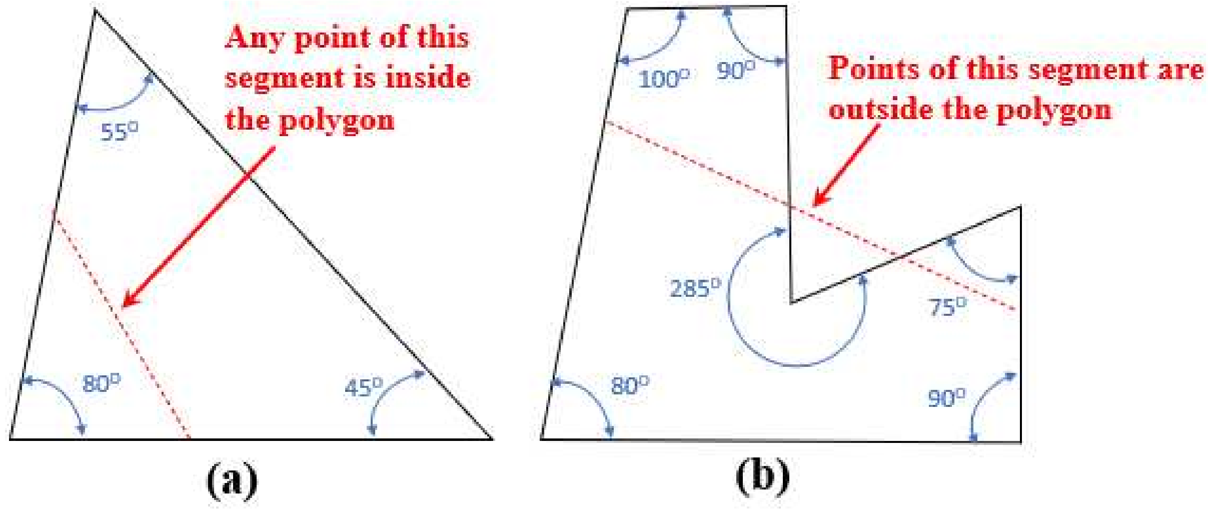

Any point of this segment is inside the polygon
55º
80º
45º
(a)
100º
90º
Points of this segment are outside the polygon
285º
75º
80º
90º
(b)

Figure 2.5 - Illustrations of the concept of (a) convex and (b) nonconvex polygons

From the current literature, there are at least three classical pattern strategies for trajectory planning applicable to WAAM, namely, raster, zigzag and contour. However, due to the geometric complexities, some of these strategies have been adapted towards layer geometry simplification (polygonal division) or merged with others (hybrid trajectory planning strategies), sharing the merits of various approaches. Figure 2.6 illustrates examples of classical and hybrid strategies adopted by different researchers. The raster strategy is likely the most known one (Figure 2.6(a)) and can be uni- or bi-directional. This becomes a flexible strategy if proper parametrisation is applied to the several starts and stops when the raster strategy is employed. Material and heat accumulation can be eliminated by setting idle times between stops and starts. An old concept used in WAAM to fill a polygon is based on a zigzagging pattern to obtain a continuous deposition per layer. As illustrated in Figure 2.6(b), with a one-way movement in a given direction up to reach a polygon border, the torch, then, faces the edge of this border for a given spacing value, before inverting the trajectory direction (in cycled pattern). Wang et al. (2019) claim process efficiency decays with this approach, due to arc extinctions that forcedly occur in parts with more complex geometries (such as internal holes). The likely first adaptation of the Zigzag strategy carried out in WAAM was proposed several years before by Dwivedi and Kovacevic (2004). This strategy was called continuous (Figure 2.6(c)), consisting of Zigzag trajectories planned to leave staggered gaps between the paths, so that the gaps were sequentially closed by another Zigzag trajectory in the inversed direction. It is noteworthy that the continuous strategy can be hybridlike adapted to other strategies, for example, spiral, as illustrated in Figure 2.6(d).

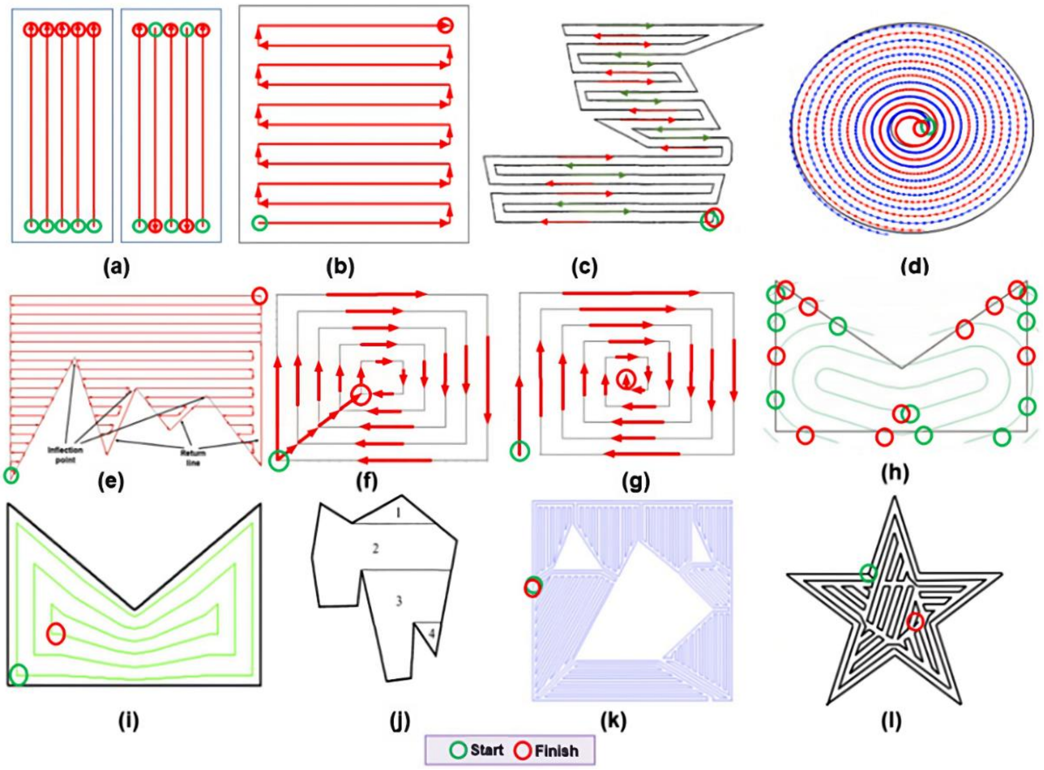


Figure 2.6 - Examples of strategies for trajectory planning in WAAM: (a) raster; (b) zigzag; (c) continuous (adapted from Dwivedi and Kovacevic, 2004); (d) continuous with the spiral contour; (e) water-pouring rule (adapted from Wang et al., 2019); (f) Parallel Contour; (g) Spiral Contour; (h) MAT, after medial axis transformation (adapted from Ding et al., 2015c); (i) adaptive MAT (adapted from Ding et al., 2016); (j) polygonal division into simpler polygons (after Dwivedi and Kovacevic, 2004); (k) convex polygonal division (adapted from Ding et al., 2014) and (l) convex polygonal division with sharp corner correction (adapted from Liu et al., 2020).

Another adaptation to the Zigzag strategy was proposed by Wang et al. (2019) and illustrated in Figure 2.6(e). The authors proposed a deposition trajectory strategy based on the water-pouring rule. In summary, the algorithm for deposition strategy is based on the identification of inflexion points (such as nonconvex angles) in the geometry of the polygon that represents layer area (see this approach described in Figure 2.6(e)). Initially, the torch follows a Zigzag pattern until an inflexion point is met. Then, the torch movement direction is reversed, so that the new shape (a partition of the polygon) after the inflexion point is filled using the same pattern. However, a return line is planned for the torch to escape from the bottom of the polygonal partition. The same procedure is maintained until all polygon partitions are filled.

In the contour strategy, Figure 2.6(f) and (g), the arc torch follows the subsequently inscribed edges of an external polygon through parallel displacements (pre-established offsets) relative to the polygon edges. The torch sweep can follow either parallel (2.6(f)) or spiral (2.6(g)) patterns, in either in-outward or out-inward orientation. However, in the case of parallel scanning, the transition between two edges is made by connecting its starting points, as indicated by red arrows in Figure 2.6(f). These contour strategies are easily applied to convex polygons, but may not perform satisfactorily when it comes to nonconvex polygons. According to Ding et al. (2015c) and Xiong et al. (2019), problems such as voids inside the parts or at very acute angled corners, and heat accumulation in the centre of the workpiece (which cause residual stress and/or deformation) are recurrent in these strategies. The out-inward scanning direction is prone to generate material accumulation at the centre unless a programmed progressive adjustment of the parameters is made.

Skeletal structures (Figure 2.6(h) and (i)) are contour adaptation approaches to solve the limitations of nonconvex shapes. Ding et al. (2015c) proposed the medium axis transformation (MAT) method for solid parts (Figure 2.6(h)), with and without holes, and for thin-walled parts. Applying MAT, they skeletonized two-dimensional geometry and performed contour-like in-outward oriented sweeps. However, this method presented noncontinuous trajectories for bulky parts, with several arc stops and restarts to avoid the torch depositing beyond the polygon edges (these interruptions are emphasized by green/red circles in Figure 2.6(h)). Ding et al. (2016) improved their approach. They presented an adaptive medial axis transformation (A-MAT), by adapting the spacing between contours around the polygon skeleton and through re-parametrisation (noticeable by gap variations between lines), resulting in continuous paths (Figure 2.6(i)).

Unlike the classical strategies presented so far, the polygon division strategy (Figure 2.6(j)) aims to reduce the complexity of nonconvex geometries by partitioning them into simpler polygons, for subsequent deposition planning, as also proposed by Dwivedi and Kovacevic (2004). According to the authors, each simple polygon is filled with one of the continuous strategies, yet in such a way that the trajectories are interconnected as a single path. However, with the increasing complexity of the initial polygon, its decomposition can result in simple but not necessarily convex polygons (see partition 2 in Figure 2.6(j), as an example). To work around this problem, Ding et al. (2014) performed a similar approach, but dividing the nonconvex polygon into only convex polygons (Figure 2.6(k)). Each sub-polygon was filled with a continuous strategy (in this case, a hybrid contour and Zigzag), interconnected as a single path. Still using the same strategy, Liu et al. (2020) segmented a nonconvex polygon into convex polygons to apply the contour and Zigzag deposition strategy, but focusing on surfaces with corners of sharp angles, as illustrated in Figure 2.6(l). In this strategy, a calculated displacement of the vertex of acute angles was applied to correct voids left during trajectory planning by contour strategy. However, this strategy of dividing polygons also presents difficulties when manufacturing parts with circular holes (WANG et al., 2019).

With a similar view to that described above, Wang et al. (2019) cite that, although there are many strategic planning strategies, these strategies can be classified into three categories, according to their origin and evolution. The first class originated in the raster method and subsequently developed into the Zigzag, continuous line and convex polygon methods. The second category originated from the contour method, which, in turn, developed into the medial axis transformation and adaptive medial axis transformation methods. The third category is the hybrid method that combines the advantages of these previous two categories by applying them in different regions.

Although often overlooked in the additive manufacturing literature (and without classification), space-filling curves such as Peano and Hilbert curves have been known since the late 19th century and are widely used in diverse applications like image processing and antenna design. In additive manufacturing, they can be referred to as Space-Filling Strategies, enabling the creation of toolpaths that fill a 2D contour entirely. Kapil et al. (2016) proposed three such strategies for WAAM, which minimized non-deposition movements; these methods can cause material accumulations that necessitate post-processing. In addition, the algorithm is complex and may be difficult to understand, as it involves multiple steps and specific cases. Recently, Singh et al. (2022) improved print quality in WAAM using Space-Filling Strategies and grid points optimized for toolpath orientation. However, the flexibility of trajectory generation to achieve multiple results may be limited. In the other hand, the results achieved by this strategies in WAAM have primarily been demonstrated on simple parts such as cubes and animals shapes, so further evidence is required to fully validate the efficacy of this strategy across a wider range of complex geometries.

2.3 The Proposal of a Novel Strategy for WAAM Deposition: The Basic-Pixel

The Basic-Pixel deposition trajectory strategy to be introduced in this work can be defined as a complex multitask procedure to carry out the trajectory planning of WAAM parts. For being the first version of the development of trajectory planning in this work, hereafter, this strategy will be referred to Basic-Pixel. The end-user, as a nonpassive agent, is required to input the height between layers, path widths, path overlapping and number of iterations and to take the final decision on the trajectory to adopt. The operation of the procedure is made through computational algorithms (heuristics), with accessible computational resources and tolerable computational time. A heuristic, or a heuristic technique, is defined in current literature as to optimization (for instance, according to Yang, 2008), as any approach aiming to find a trial-and-error solution that may not be optimal but is sufficient (acceptable), considering a reasonably practical timeframe. In a few words, the model layers in Basic-Pixel are fractioned in squared grids, over which the trajectory is planned. To model the problem, a set of dots is generated and distributed within the top-view cross-section outline of the slice, resembling pixels on a screen (a dot matrix to form a raster graphic, in other words, a grid composed of a collection of small squares). The useful intersections between the grid lines and the edges of the slice are hereafter referred to as nodes. The technique used in this Basic-Pixel strategy for optimization was based on creating a trajectory from a well-known route optimization named Travelling Salesman Problem (TSP), whose challenge is to find the shortest yet most efficient route for the torch to take, given a list of specific positions. To simplify the algorithm and eliminate the need for tuning multiple parameters, the Basic-Pixel strategy employs an adapted version of the Greedy Randomized Adaptive Search Procedure (GRASP) metaheuristic for additive manufacturing. GRASP is aided by four distinct concurrent heuristics of trajectory planning, namely nearest neighbour, biased, alternate and random contour, and the 2-opt algorithm. This approach consists of iterations made up of successive constructions of randomized initial solutions (global search) and subsequent iterative improvements (local search). After all the recurrent loops, a trajectory is defined and written in machine code.

To detail the proposal, the Basic-Pixel strategy itself is flow charted as in Figure 2.7. A basic process chain for WAAM begins with a 3D CAD model that is converted into an AMF or STL format file, which is accordingly sliced by dedicated software, hereafter alluded to as the "printing process planning software" (PPPS). From a PPPS is expected more than the basic functions of slicing the model and generating a machine code for the WAAM printer. In general, before machine code generation, a proper PPPS should also define the tool path (trajectory) planning. The first steps of the process, i.e., reading the STL file from the 3D model, orientation optimization and introduction of the layer heights and digital slicing of the model, are usually common, yet comprehensive, printing process planning. They will not be discussed further here, keeping the arguments concentrated on the following four steps of the proposed Basic-Pixel torch trajectory planning:

1. Discretization of the layers (through the distribution of dots all over the layer surfaces, i.e., modelling the environment as a grid);
2. Starting position definition and node connections (generating an initial trajectory);

3. Trajectory Optimization (recurrent start position choices and reconnections of the nodes from the trajectory generated in the previous step, in order to obtain the shortest path to be travelled by the torch);

4. Storage of the generated optimized trajectories (to compose a set of batches of trajectories to be selected and, accordingly, supply instructions to the printing machine, in the form of coordinates in an array).

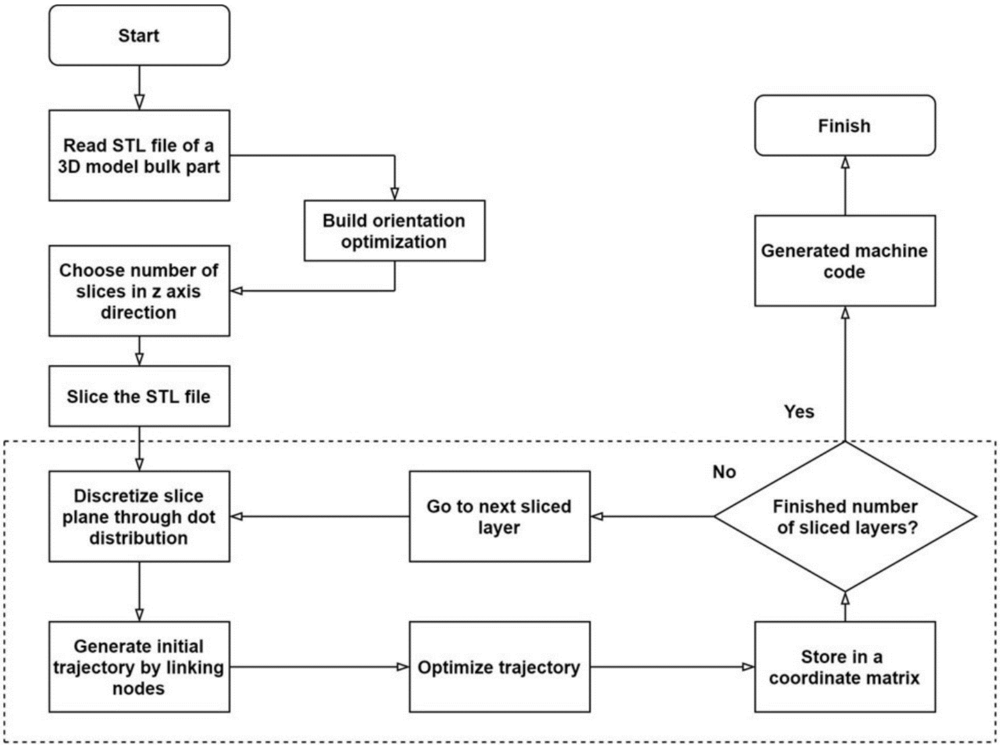

Start
Read STL file of a 3D model bulk part
Build orientation optimization
Choose number of slices in z axis direction
Slice the STL file
Discretize slice plane through dot distribution
Go to next sliced layer
No
Finished number of sliced layers?
Yes
Generate initial trajectory by linking nodes
Optimize trajectory
Store in a coordinate matrix
Generated machine code
Finish

Figure 2.7 - A basic process chain for WAAM, highlighting the workflow of the Basic-Pixel strategy for trajectory planning (within the dashed lines)

These four steps are repeated over all layers generated by the slicing process (in the case of layers with the same cross-section, a trajectory generated in the initial layer can be replicated in the others). After concluding the loop, the best (according to the criterion) stored coordinates array is converted into machine code, which is sent to the WAAM printer. It's important to note that steps 2, 3, and 4 correspond to the application of the GRASP metaheuristic, which will be discussed in detail in the following section. The entire algorithm described here has been implemented in the open-source software Scilab.

2.3.1 Discretization of Layers and Node Indexing

The discretization process, as proposed here, consists of four phases, that is, insertion of offset contours in the sliced layers, generation of a grid to generate dots (these two first stages are demonstrated in Figure 2.8), simplification of dots (as illustrated in Figure 2.9) and indexing of nodes (as established in Figure 2.10). In the first phase, the process provides a new contour to the original polygon (Figure 2.8(a)) at an offset distance from the slice edges, represented by the letter "v" in Figure 2.8(b), composed of a simplified shape (for example purposes). From this equidistance, an internal (inscribed) surface contour is created to reduce excess or avoid material shortages at the edges of the original layer shape (for other geometries rather than that in Figure 2.8(a), the offset would take another profile, yet keeping the same role). Then, in a second phase, equally spaced horizontal and vertical line segments (from the lower and most left-positioned vertex of the offset contour) are virtually plotted over the plane, performing a grid. The intersections of these lines (blue dots in Figure 2.8(c)) and between these lines and the edges of the offset (green dots in Figure 2.8(d)) and the offset contour vertices (red dots in Figure 2.8(b)), all together form the pixel dots. It is emphasized that the spacing value is an input to the algorithm in question, to be defined by the process analyst, and should be considered as the distance between two weld beads arranged side by side (considering potential overlapping). It is therefore justified that equidistance may not be obtained at the top and right edges of the offset contour.

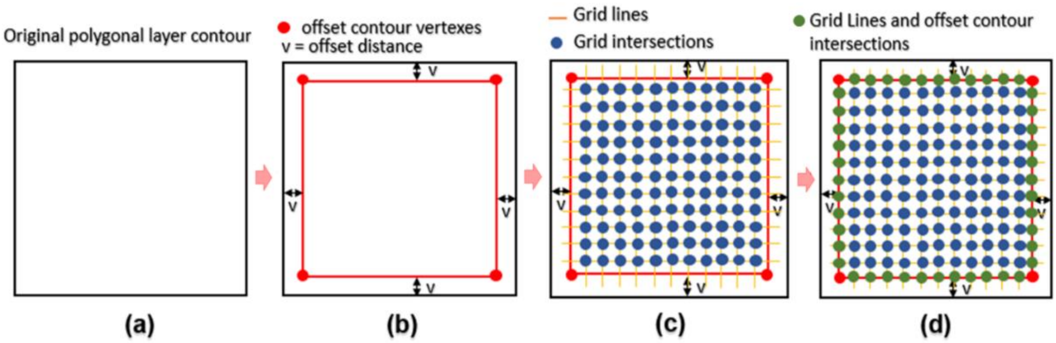

Original polygonal layer contour
● offset contour vertexes
v = offset distance
— Grid lines
● Grid intersections
● Grid Lines and offset contour intersections
v
(a)
(b)
(c)
(d)

Figure 2.8 - Discretization of a layer surface according to the Basic-Pixel strategy: (a) original polygonal layer contour; (b) insert of the inscribed offset contour and respective generation of red dots; (c) generation of blue dots; and (d) generation of green dots

It is important to emphasize that coincident dots may arise, due to layer topology and the spacing values of the segments, as illustrated in Figure 2.9. In this example, the coincident dot arose because a straight segment (in yellow) of the grid crosses the offset contour (in red) at one of the vertices, which is also taken as an intersection of the discretization. There are cases where dots are coincident, or almost coincident, also highlighted in Figure 2.9. Therefore, a simplification of the model should be applied. Only one of these overlapping dots is considered and the others are eliminated from the discretization, which reduces the processing time to generate the trajectory. In addition, the model eliminates nearby dots that are separated by a specific Euclidean distance that is usually set to 30% of the bead width, as these dots do not significantly contribute to the deposition process. It is worth mentioning that the percentage value used to simplification of dots can be adjusted by the user. Considering the elimination of dots created during the initial discretization, the remaining dots (useful dots) are named nodes.

Given continuity to the discretization protocol, the assigned nodes are ordered from left to right and bottom to top and, following, indexed (coded) numerically in ascending order. This indexing of each node is already illustrated in Figure 2.10, where $\boldsymbol{i_1}$ represents the first node of the layer (with a respective coordinate). Accordingly, the following nodes $\boldsymbol{i_n}$ the same horizontal row are indexed in ascending order, arranged to the right, i.e., dot $\boldsymbol{i_n}$ represents the last node of the first horizontal layer row and dot $\boldsymbol{i_{n+1}}$ represents the first node of the second horizontal layer row. Hence, the indexing resumes at the second row from the first column, following the same pattern until the last dot ($\boldsymbol{i_{max}}$) is reached at the rightmost point of the utmost row.

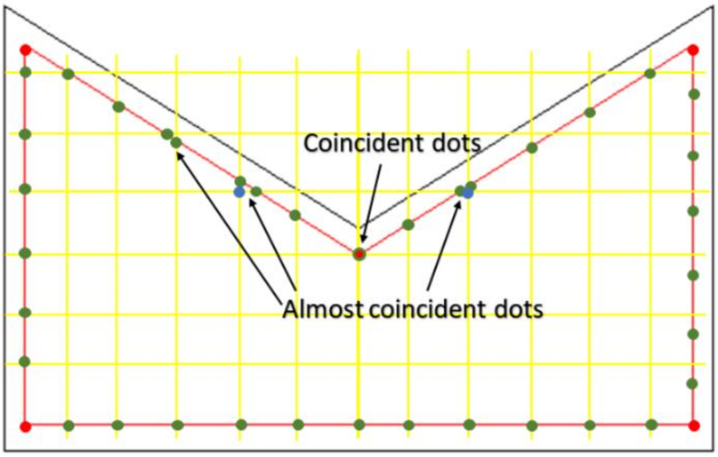


Figure 2.9 - Coincident and almost coincident dots during the discretization stage of a layer

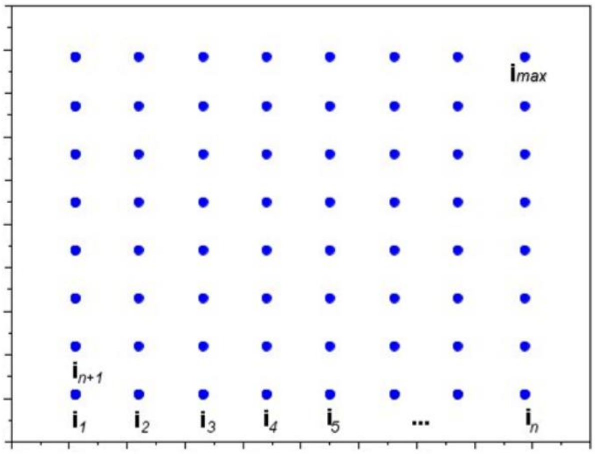
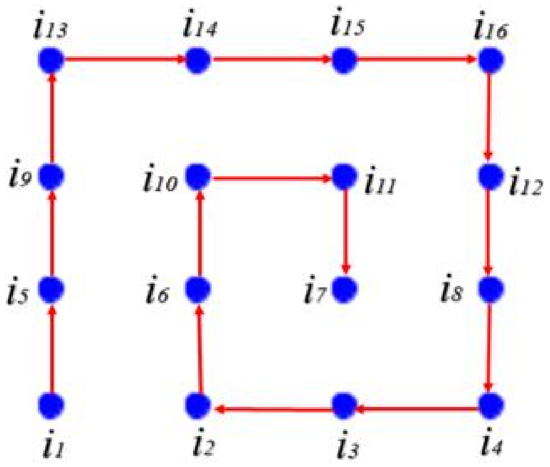


Figure 2.10 - Indexing of nodes by ordering intersected dots distributed on a layer surface

2.3.2 Starting Position Definition and Node Connections to Define a Trajectory

The sets of nodes and their connections were modelled according to the Travelling Salesman Problem (TSP). This approach is based on the problem of a traveling salesman who leaves an initial city and must visit all the cities programmed in a row and return to the city of departure. As a goal, he/she should plan the shortest route, however not going to a city already visited. In general, two solutions for the TSP are known. In the first, a start node is defined at random by a computer routine and a first route is generated, applying the rule of no-duplicated visits, followed by other routes improved by local search heuristics. The second solution would be to create alternative routes from different start nodes. By one means or the other (or even together), the best route is eventually provided. Ouaarab (2020) and Zia et al. (2018) showed the different approaches to solve the TSP in specific cases, such as railway track optimization, robot movement, vehicle routing, among others. Wasser et al. (1999), in turn, successfully applied heuristic of trajectory planning in additive manufacturing of polymers.

To simulate this problem as a base for the Basic-Pixel strategy, each node (the remaining dots) created on the surface (Section 2.3.1) corresponds to a city of the TSP and the node connections resemble the path travelled by the salesmen, i.e., the deposition trajectory. Figure 2.11 is a diagrammatic representation of this problem, in which the set of interconnected nodes is represented by i_1, i_2, i_3, …, i_{16}, where $i_1 \neq i_2 \neq i_3 \neq \ldots \neq i_{16}$ for the case in which 16 is the total number of nodes of this example. A set of trajectories (T) is represented by defined connections between nodes, each node with one entrance and one exit, where $\overrightarrow{i_1 i_5}$, for instance, represents an effective link between node i_1 and node i_5. Note that in this representation, the last visited “city” (i_7) and the initial visited “city” (i_1) are not connected (even the first node is not revisited).



$$\{\overrightarrow{i_1 i_5}, \overrightarrow{i_5 i_9}, \overrightarrow{i_9 i_{13}}, \overrightarrow{i_{13} i_{14}}, \overrightarrow{i_{14} i_{15}}, \overrightarrow{i_{15} i_{16}}, \overrightarrow{i_{16} i_{12}}, \overrightarrow{i_{12} i_4}, \overrightarrow{i_4 i_3}, \overrightarrow{i_3 i_2}, \overrightarrow{i_2 i_6}, \overrightarrow{i_6 i_{10}}, \overrightarrow{i_{10} i_{11}}, \overrightarrow{i_{11} i_7}\} \in T$$

Figure 2.11 - Diagram using Travelling Salesman Problem (TSP) to define a hypothetical trajectory as done in the Basic-Pixel strategy for WAAM (observed that each node has a single entrance and exit)

The distance between two *nodes* i_p and i_q, where *p* is a given node and *q* is a subsequent node, is calculated by the vector distance between the two coordinates (i_{px}, i_{py}) and ($i_{qx,}, i_{qy}$),, as expressed in Eq. 2.1, which represents the Euclidean distance (*d*).

$$d(i_p i_q) = \sqrt{\left(i_{px} - i_{qx}\right)^2 + \left(i_{py} - i_{qy}\right)^2} \tag{2.1}$$

The total trajectory distance (D_T) is calculated according to the summation of all defined connections between two nodes ∈ *T*, as described in Eq. 2.2:

$$D_T = \sum_{f=1}^{n} d_f(i_p i_q) \tag{2.2}$$

where *n* represents the total number of connections between nodes that belongs to *T,* and *f* represents the indexes of each connection.

In the Basic-Pixel trajectory deposition strategy, different heuristics of trajectory planning were used for sequential node connections (sometimes referred to as heuristics for simplification) following the TSP principle. The four heuristics employed in the current work are denominated Nearest Neighbour, Biased, Alternate and Random Contour, and they are described in the next subsections. The goal was to force, with each of the heuristics, so as to have assessed by the algorithms some of the strategies shown in Fig. 2.6, yet under the concept of the Basic-Pixel strategy. For implementing the heuristics in the Basic-Pixel strategy, a computer code was developed.

2.3.2.1 Nearest Neighbour Heuristic for trajectory planning

According to Ouaarab (2020), the Nearest Neighbour Heuristic is widely used for TSP solutions, and for this reason, this heuristic was also applied in the Basic-Pixel strategy. Figure 2.12 shows a flowchart to schematize this heuristic for trajectory planning. The central idea is to select (as mentioned before, by raffling) an initial node to be visited (starting node) in any region of the discretized layer and from that to proceed to the shortest distance node (next node to be activated). The proximity criterion is based on the Euclidean distance, already presented in Eq. 2.1. However, as the nodes generated on the surface of the layer are usually equidistant during the selection of the nearest node (Figure 2.13), more than one candidate with the same Euclidean distance is prone to appear. Among the candidate nodes, the algorithm will choose one at random and take it as the next activated node. The, until then, an active node is reassigned as a visited node. In sequence, according to the proximity criterion, the cycle is repeated until there are no more unvisited nodes, and a trajectory is generated by connecting all discretized nodes in the layer one by one.

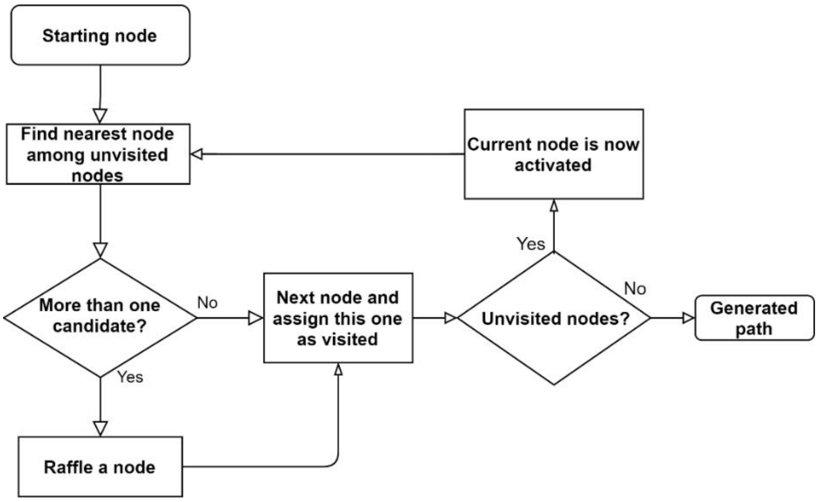

Starting node
Find nearest node among unvisited nodes
More than one candidate?
No
Yes
Raffle a node
Next node and assign this one as visited
Unvisited nodes?
Yes
No
Current node is now activated
Generated path

Figure 2.12 - Nearest Neighbour heuristic for trajectory planning flowchart

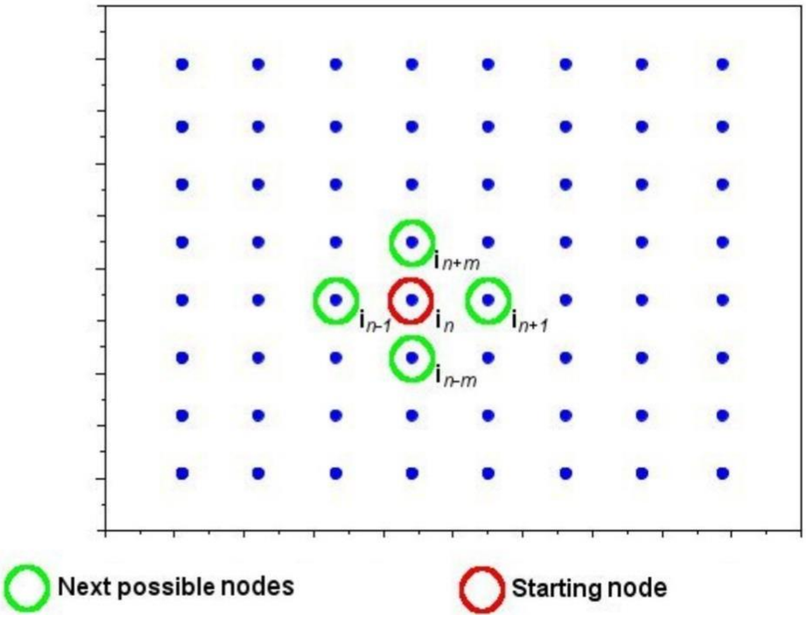


Figure 2.13 - Index representation of a target node and its nearest neighbours

2.3.2.2 Biased heuristic for trajectory planning

Differently from the Nearest Neighbour heuristic, which was already known from the literature, the Biased Heuristic was created for this work to force a Zigzag deposition trajectory strategy. Figure 2.14 shows the flowchart of the Biased heuristic, which is similar to the previous one, although presenting a difference in the decision-making when there is more than one equidistant candidate closer to the active node, i.e., instead of a random choice, there is a second scrutiny, which is based on the index value instead of the Euclidian distance. In this case, the node with the nearest index value is flagged and, if there is more than one non-visited candidate, the heuristic will pick up the one with the succeeding index value. For example, in Figure 2.13, the next node to be activated after $\boldsymbol{i_n}$ would be $\boldsymbol{i_{n+1}}$, despite the fact that there would be four equidistant nodes ($\boldsymbol{i_{n-1}}$; $\boldsymbol{i_{n-m}}$; $\boldsymbol{i_{n+1}}$; $\boldsymbol{i_{n+m}}$) and two preceding and succeeding index value nodes ($\boldsymbol{i_{n-1}}$; $\boldsymbol{i_{n+1}}$).

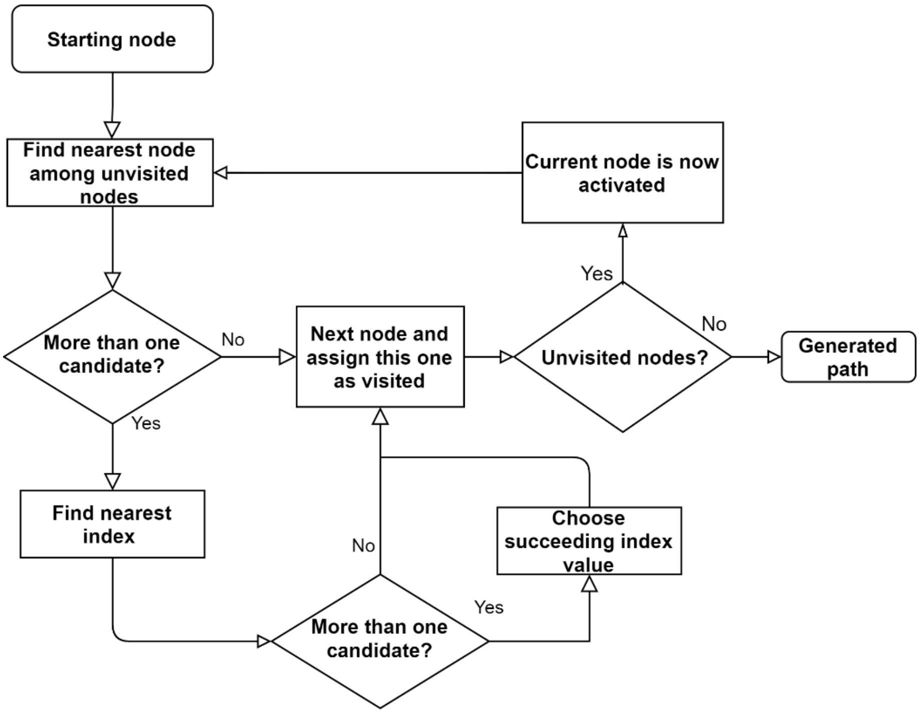

Starting node
Find nearest node among unvisited nodes
Current node is now activated
Yes
More than one candidate?
No
Next node and assign this one as visited
Unvisited nodes?
No
Generated path
Yes
Find nearest index
No
Choose succeeding index value
Yes
More than one candidate?

Figure 2.14 - Biased heuristic flowchart for trajectory planning

2.3.2.3 Alternate Heuristic for trajectory planning

The Alternate heuristic flowchart is presented in Figure 2.15. Like the Biased heuristic, the Alternate heuristic was created explicitly for the Basic-Pixel strategy, aiming at imposing torch oscillation during deposition, a technique commonly used in welding. This heuristic name comes from the fact that the decision is to do depositions alternately when there is more than one equidistant candidate to the next node. In this case, if the routine iteration takes an odd value, the node to be activated will be the node with the index value that presents a higher difference to the active index node. Having more than one candidate, the candidate with the smaller index value will be activated. Alternatively, in the case of even iterations, the node to be activated will be the node with the index value that presents the shortest difference to the active index node. In the case of more than one candidate, the one with the longest index value will be activated. Using Figure 2.13 again as an example, at first iteration, for which x = 0, the next node to be activated would be $\boldsymbol{i_{n-m}}$, because it has a greater index value difference from $\boldsymbol{i_n}$ and it is the one with a smaller index value between $\boldsymbol{i_{n-m}}$ and $\boldsymbol{i_{n+m}}$. The subsequent node after $\boldsymbol{i_{n-m}}$ would be $\boldsymbol{i_{n-m+1}}$.

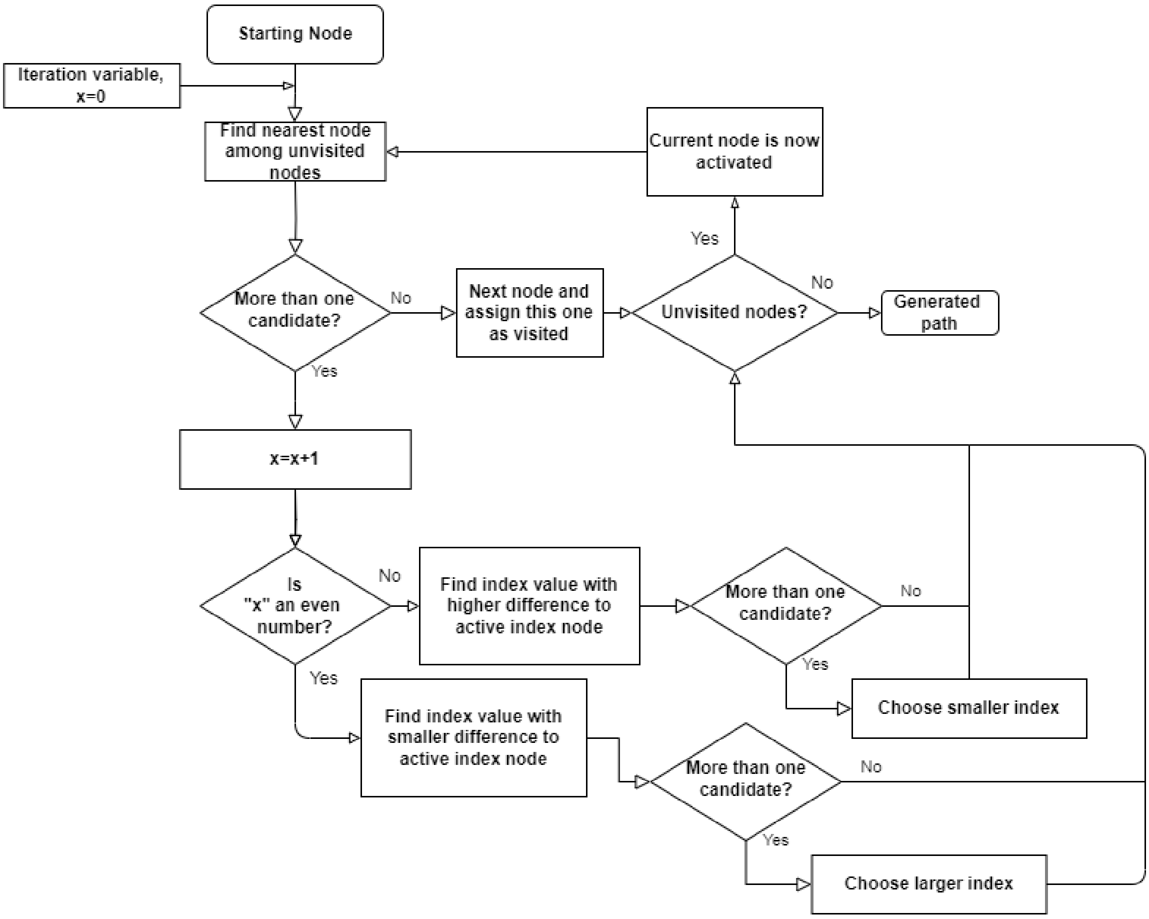

Starting Node
Iteration variable, x=0
Find nearest node among unvisited nodes
Current node is now activated
Yes
More than one candidate?
No
Next node and assign this one as visited
Unvisited nodes?
No
Generated path
Yes
x=x+1
Is "x" an even number?
No
Find index value with higher difference to active index node
More than one candidate?
No
Yes
Choose smaller index
Yes
Find index value with smaller difference to active index node
More than one candidate?
No
Yes
Choose larger index

Figure 2.15 - Alternate heuristic flowchart for trajectory planning

2.3.2.4 Random Contour heuristic for trajectory planning

The Random Contour heuristic is another heuristic created for the Basic-Pixel trajectory planning to pursue the contour deposition trajectory strategy. According to its flowchart presented in Figure 2.16, when a draw happens between the closest candidates (equidistance criterion), decision-making is made in favour of the node that is closest to the polygon edges (internal and external). In the case that there is still a draw, the decision is made by a raffle between the drawn nodes. To exemplify this latter case using Figure 2.13 again, the nodes $\boldsymbol{i_{n-m}}$ (vertically) and $\boldsymbol{i_{n-1}}$

(horizontally) would be equidistant to ***i**$_n$*, but equivalent as nodes closest to the polygon edges, taking into account all directions. In this case, a choice at random would be made to define one of the two alternatives.

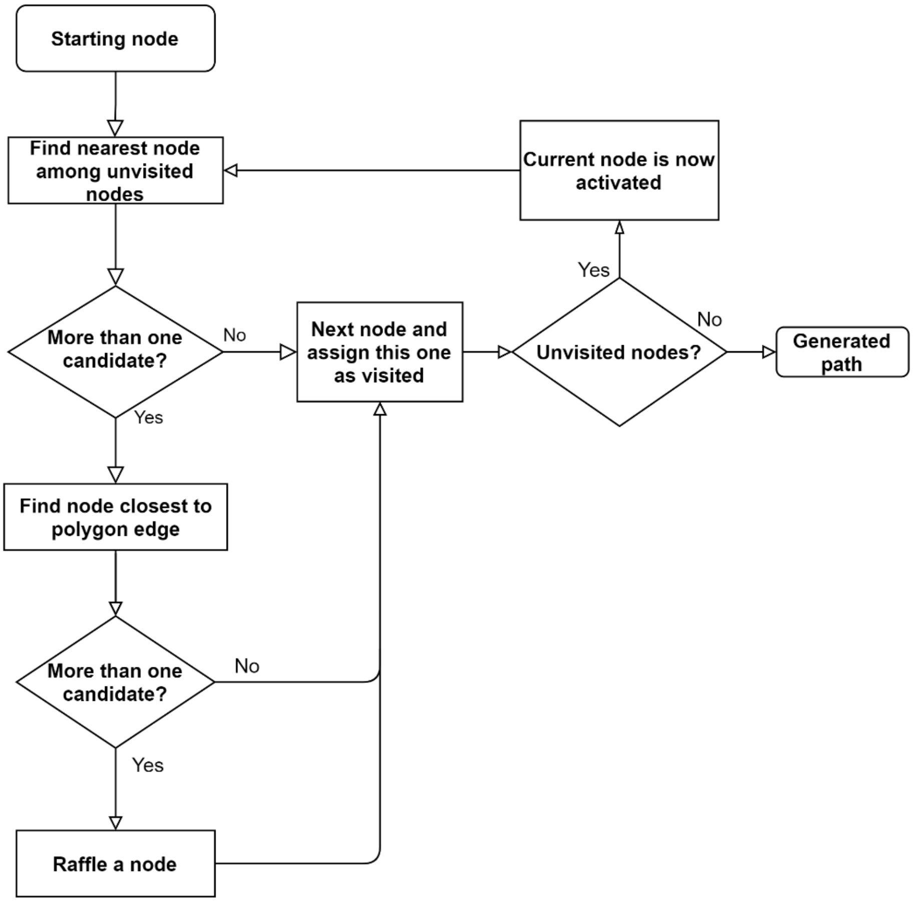

Starting node
Find nearest node among unvisited nodes
Current node is now activated
Yes
No
More than one candidate?
No
Next node and assign this one as visited
Unvisited nodes?
Generated path
Yes
Find node closest to polygon edge
More than one candidate?
No
Yes
Raffle a node

Figure 2.16 - Random Contour heuristic flowchart for trajectory planning

2.3.2.5 Examples of trajectories generated by the 4 heuristics

For a better visualization of the different heuristics of trajectory planning applied by the Basic-Pixel strategy, Figure 2.17 illustrates their application on a polygonal square-shaped surface. By way of example, all heuristics have taken node 1 as a starting node. Figure 2.17(a) illustrates the trajectory that the nearest neighbour heuristic generates. The trajectory generated by the Biased heuristic (Figure 2.17(b)), in turn, is very similar to the Zigzag strategy, but modelled by dots rather than segments. In Figure 2.17(c), where the algorithm tested the Alternate heuristic, the trajectory is typically a square wave-shaped oscillation. Finally, in Figure 2.17(d) is shown how the Random

Contour heuristic would present the trajectory, similar to the Spiral-Contour strategy.

As seen, the objectives of each of the heuristics were reached. Naturally, with a random choice of the starting node and recursively running the heuristic algorithms, different trajectory patterns could result. The total trajectory distances (D_T), used for decision-making among the delivered options, are, and would be, undoubtedly different.

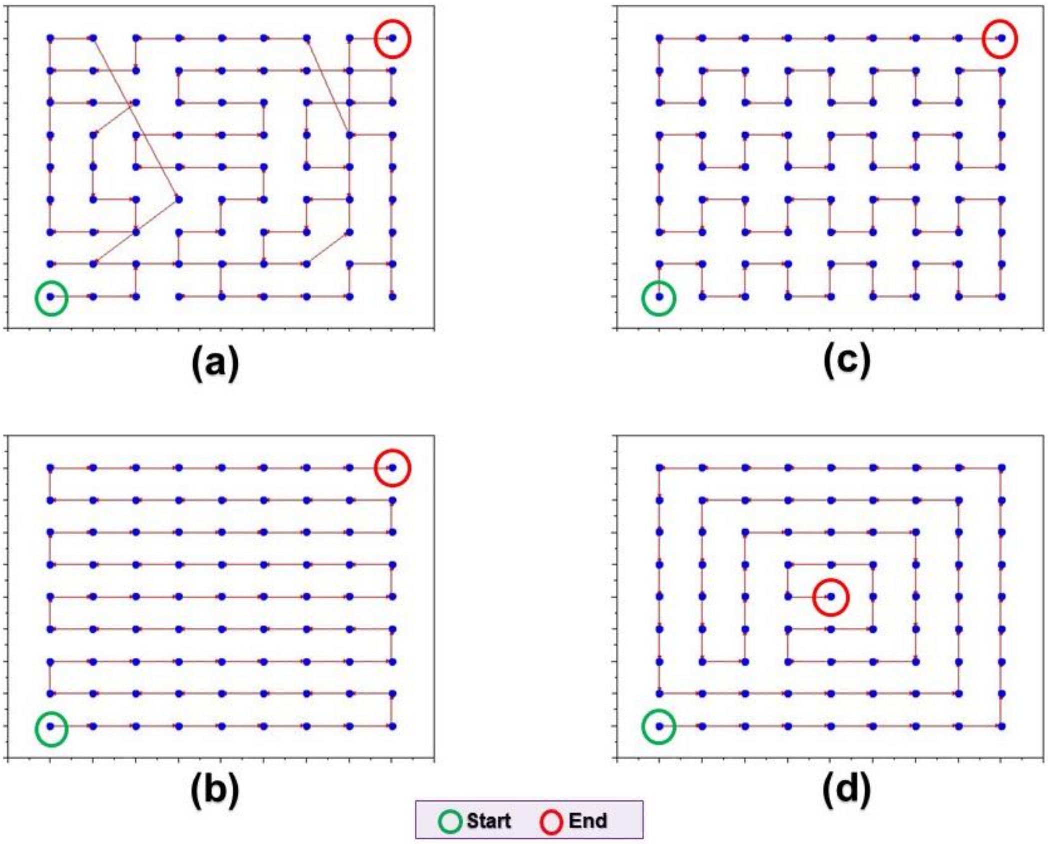


Figure 2.17 - Trajectories generated by the heuristic with a single iteration and using the same starting node: (a) Nearest Neighbour; (b) Biased; (c) Alternate; (d) Random Contour

2.3.3 Trajectory Optimization

Trajectories in WAAM should preferably be continuous, or with minimal interruptions, to reduce manufacturing time and improve the surface homogeneity of the printed parts (the shorter the route, the better the objective function of the optimization). For this, the paths generated (node connections to define a trajectory) in the previous subsections must always connect neighbouring nodes, yet never pass through nodes already “visited” or present path crossing (which will result in the accumulation of material in these regions). In Figure 2.17(a), for example, the Nearest Neighbour heuristic presented path crossings, which showed the need for improvement. In the other heuristics, although they did not present path crossings, the algorithms could deliver trajectories with path crossings as a function of other starting nodes and/or higher geometry complexity. Paths with no deposition (“empty” trajectories) to avoid path crossing, in turn, can be adopted, however, at the expense of arc stops and starts (regions predisposed to imperfections) and potentially increased manufacturing time. Therefore, penalties are inserted in the algorithm when path crossings (and other constraints) are identified (first optimization constraint factor).

To perform optimization for the objective function, Equation (2.2), and constraints in the Basic-Pixel strategy, a local search through the 2-opt algorithm, commonly used as heuristics of improvements for TSP, as can be seen in the study by Lee et al. (2020), was initially applied. This improvement consisted of initially choosing two paths between the nodes of the generated trajectory and reversing their connections to verify if a reduction in path distance would occur. Figure 2.18(a) presents an example where path 1, initially chosen by one sequential node connection heuristic, was represented by node A interconnected to node B and path 2 represented by node C interconnected to node D. When applying the 2-opt algorithm, the chosen paths were undone and uncrossed, that is, node A now generated a path connecting to node C, and node B generated a path interconnected to node D, as illustrated in Figure 2.18(b). If there is a reduction in the Euclidean distance, the new solution is adopted. Otherwise, the swap is undone, and another pair of neighbouring nodes is chosen and the process is repeated over and over again. This operation is repeated until no further improvements occur with path crossings. With this, a great local solution is obtained.

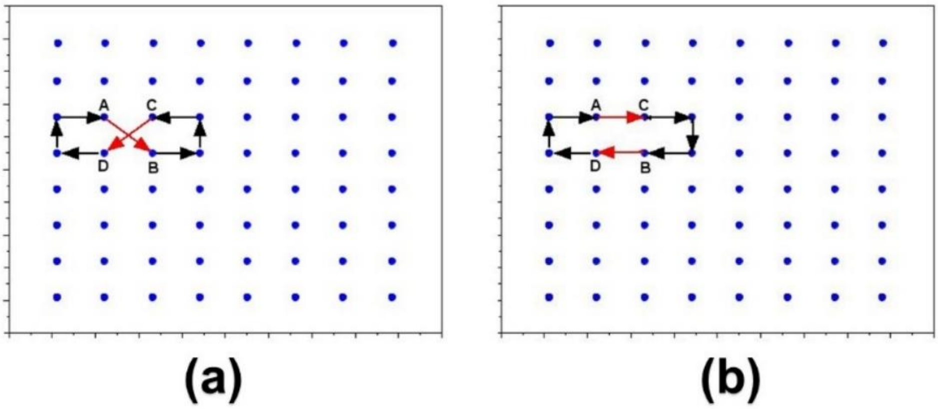


Figure 2.18 - 2-opt method for local search optimization in the Basic-Pixel trajectory strategy: (a) initial trajectory; and (b) trajectory optimized by uncrossing two paths

Despite the good results obtained by the 2-opt heuristic, this technique is restricted to a maximum or minimum local only (such as entering a valley or climbing a peak of the search space). In order to allow exploration of other valleys/peaks in the space, the local search is boosted by previously using the metaheuristic GRASP, which consists of a higher-level heuristic procedure designed to find a better solution to an optimization problem. GRASP basically consists of a multiboot iterative technique for global search from successively constructed randomized solutions and subsequent iterative improvements. According to Sohrabi et al. (2020), each iteration performs two perfectly defined phases. According to these authors, the first phase creates viable solutions to the problem, which promotes diversity, and the second phase consists of the optimization from the created solutions.

Figure 2.19 shows the GRASP flowchart adapted to the Basic-Pixel strategy for WAAM, which starts with inputting the number of iterations from the process analyst. After a random selection for picking up the starting node, the algorithm performs the trajectory planning using the four heuristics presented in the previous section, that is, the Nearest Beighbor heuristic, the Biased heuristic, the Alternate heuristic and the Random Contour heuristic. After that, each of them is optimized with the 2-opt algorithm, which will result in four values of local minima. The lowest value obtained (shortest route, unless defined by another objective function) is stored in an array of best values. Then, a subsequent iteration is performed until the number of iterations is zeroed out. In this application of GRASP, the loop resumes from the random selection of a new starting node, which allows diversity in the initialization phase and a global exploration of the search space.

One outstanding point of this adaptation is that GRASP has an array of better values and related metrics as outputs. At first, the process analyst chooses the trajectory that will print the part, usually based on the shortest distance, already avoiding the path crossings and node visit duplication. To monitor the functionality of the Basic-Pixel strategy, the algorithm delivers a report that relates the trajectories of the best value matrix to its heuristics of trajectory planning. With this, it is possible to check if one or more heuristics of trajectory planning tend to always generate the shortest routes (a self-learning evaluation for the process analyst).

This strategy is useful for the process analyst to make decisions systematically, considering that the algorithm can be executed with all four heuristic of trajectory planning, or only with the selected ones. The latter option reduces the computational time to generate the trajectories. As another alternative to reduce computational time, the algorithm's stop criterion is optionally added for when the first trajectory without path crossings is found, as it would already be a viable trajectory for WAAM.

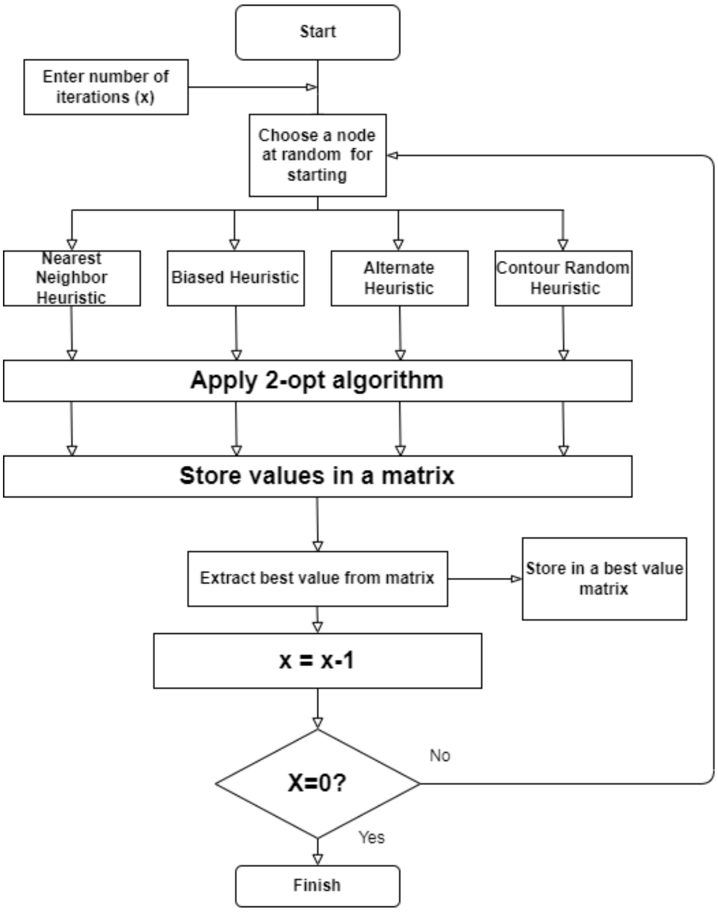

Start
Enter number of iterations (x)
Choose a node at random for starting
Nearest Neighbor Heuristic
Biased Heuristic
Alternate Heuristic
Contour Random Heuristic
Apply 2-opt algorithm
Store values in a matrix
Extract best value from matrix
Store in a best value matrix
x = x-1
X=0?
No
Yes
Finish

Figure 2.19 - GRASP metaheuristic flowchart used for the global search in the Basic-Pixel trajectory planning strategy

2.4 Computational Evaluation of the Basic-Pixel Path Planning Strategy

The aim of this computational evaluation is to showcase the flexibility of the Basic-Pixel strategy in generating multiple trajectories using all the available heuristics of trajectory planning. This approach demonstrates the ability to produce multiple printing alternatives in each iteration. For this evaluation, a slice in the shape of a nonconvex polygon (Figure 2.20(a)) was chosen as the case study for the computational evaluation of the Basic-Pixel trajectory strategy approach. This geometry symbolized a problematic condition for WAAM, yet not too complex to be analyzed and explained here. First, the part was sliced (Figure 2.20(b)) with a given increment (in this case, 2.2 mm). Figure 2.20(c) displays the top view of one of the layers to be treated by the Basic-Pixel strategy. Starting with the discretization of the layer, an initial internal offset of a given distance (in this case, 4.5 mm) from the polygon edges was implemented (Figure 2.20(d)). The spacing between the dots was set (in this case, 6 mm), with the grid starting from the leftmost and lower sides of the offset (Figure 2.20(e)), totalling 303 nodes generated on the surface. The number of nodes depend on the spacing (given by the user) and plane area. Finally, the adapted GRASP metaheuristic procedure (encompassing the 2-opt local search algorithm) was used to generate optimized trajectories. Seventy-six iterations were used, equivalent to 25 % of the total number of nodes (this percentage was arbitrarily chosen). As a result, a matrix of best values was fed with the 76 optimized trajectories. Figure 2.20(f) illustrates the shortest trajectory distance reached by the Random Contour heuristic.

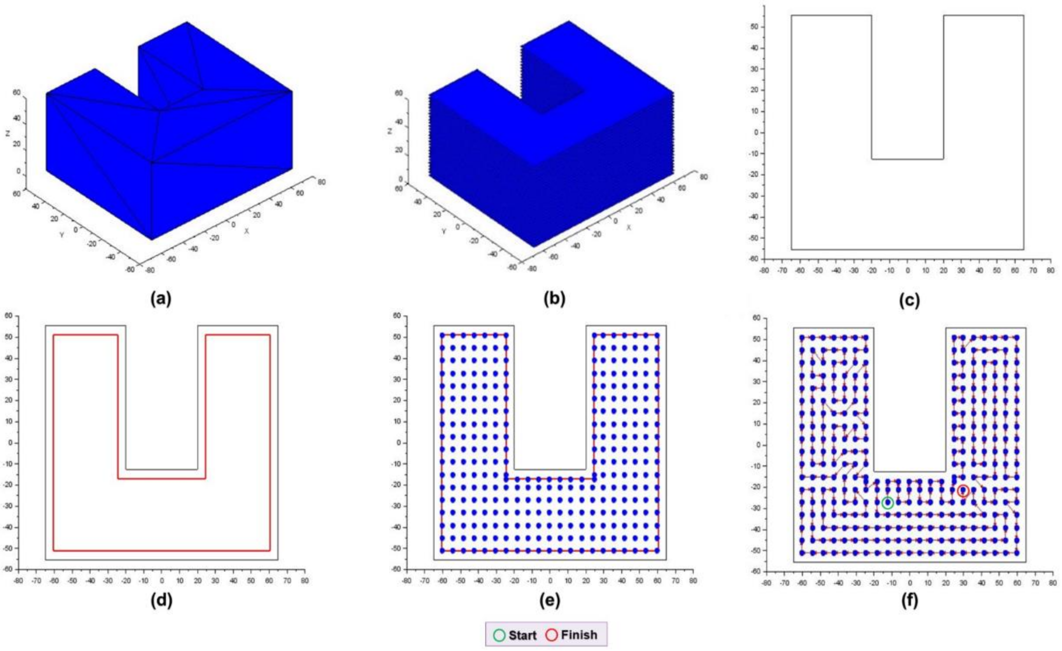


Figure 2.20 - Sequencing of the Basic-Pixel strategy used for trajectory planning of a case study (a nonconvex prismatic shape): (a) 3D CAD model (exportable as. stl format); (b) sliced model; (c) top view of polygonal layer (slice); (d) offset lines (4.5 mm edges) inscribed in the polygonal layer; (e) nodes generated inside polygonal layer; and (f) shortest trajectory generated after 76 iterations reached by the Random Contour heuristic

Figure 2.21 presents the matrix of the best values in graphical form. Each point represented the best trajectory (in terms of distance) of an iteration. In the abscissa, the four heuristics of trajectory planning were indicated, so that one could see the number of shortest distances per iteration resultant from each heuristic, as well as the longest and shortest distances after all iterations. It can be seen that the Random Contour heuristic was the most efficient, with the highest number (28) of shortest distances after 76 iterations, against 24 of the Nearest Neighbor, 13 of Alternate and 11 of Biased. In the study case, the Random Contour heuristic was that which also produced the shortest trajectory among the whole array, with 1817.72 mm, followed closely by the Nearest Neighbour heuristic, with 1819.61 mm. On the other hand, the same Random Contour also produced the second longest trajectory among the shortest trajectory per iteration. This reinforces the casual character of the process. Therefore, it is important to understand that both other geometry and a different number of iterations (or even a new run of the GRASP algorithm), could deliver other results concerning the most efficient heuristic of trajectory planning.

However, while the Random Contour heuristic produced the lowest trajectory value, the other heuristics also generated continuous trajectories that are feasible for WAAM of this prismatic shape. It is worth noting that in addition to the commonly used Nearest Neighbour heuristic in TSP, other heuristics of trajectory planning also produced good results, demonstrating the flexibility of the Basic-Pixel strategy. Compared to different Space-Filling strategies, such as the Hilbert strategy, the Basic-Pixel strategy creates a trajectory without constant changes in direction. This not only can lead to better dimensional quality, but also distributes the trajectory evenly, potentially improving temperature distribution and reducing residual stress, as demonstrated by Sun et al. (2021).

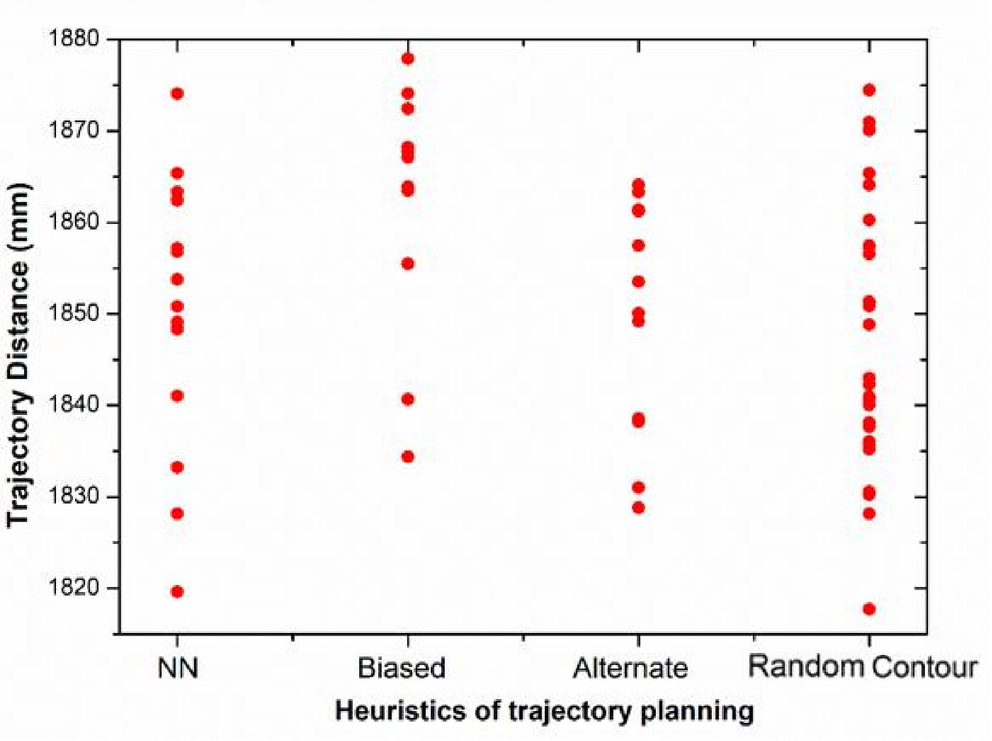


Figure 2.21 - Distances of result trajectories after each of the 76 GRASP iterations, as a function of the heuristics of trajectory planning (NN – Nearest Neighbour), where each dot corresponds to one iteration

2.5 Experimental Evaluation of the Basic-Pixel Path Planning Strategy

2.5.1 Methodology

This experimental evaluation aimed at validating the effectiveness of the Basic-Pixel strategy in the context of WAAM technology. The quality of the printed parts was assessed by comparing them with the original 3D model. For digitization, the 3D HandyScan 3D$^{\text{TM scanner}}$ from Creaform was used. The meshes obtained from scanning were compared with the 3D model surface using the VXElements software from the same scanner company. To identify any internal discontinuity, the printed parts were cut longitudinally.

As case studies, two geometries, illustrated in Figure 2.22, were manufactured using the trajectory planning developed with the Basic-Pixel strategy. The manufacturing process was carried out using a CNC gantry specifically designed for WAAM, which controlled the torch's X-Y-Z movements. The deposition was carried out using Fronius CMT equipment (Fronius International GMBH, Wels, Austria) over a substrate made of a SAE 1020 steel plate, with dimensions of 300 mm × 180 mm × 12 mm. The substrate was cooled with water (immersed), characterizing passive thermal management. Further details of the experimental rig can be found in Silva et al. (2020). A 1.2-mm-diameter AWS ER70S-6 class wire was employed, with the arc shielded by a blend of Ar and CO_2 (4 %). The contact tip-to-work distance was set at 12 mm. The deposition and wire feed speeds were 32 cm/min and 4.1 m/min, respectively, leading to 2.8-mm-high and 4.1 mm wide straight paths.

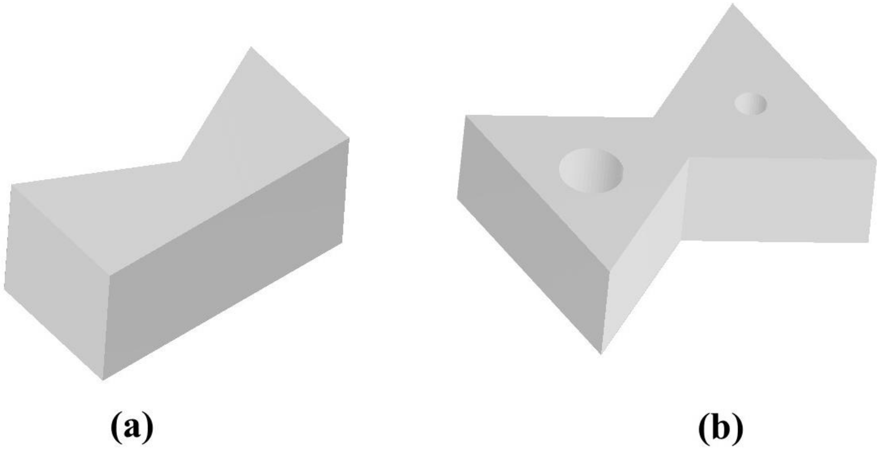


Figure 2.22 - 3D Printed Models Using Basic-Pixel Strategy: (a) Ding et al. (2016) shape; and (b) Wang et al. (2019) shape

Therefore, the inputs to the program to perform the Basic-Pixel strategy were an offset distance of 2 mm (half-width of the deposited paths) and the distance between the nodes was fixed at 3.03 mm, considering a 73.8 % of path overlapping of the bead width as suggested by Ding et al. (2015d). It is noteworthy that parameter optimization for printing was not adopted, because it would be out of the work scope, even though the authors recognize that proper parameterization is crucial to reach sound trajectories. These deposition parameters and path planning were used for both geometries.

2.5.2 Results and Discussions

Figure 2.23 presents the prismatic shape of the first geometry, the trajectory planned by using the Basic-Pixel strategy (utilizing 383 nodes, after 50 iterations with the four heuristics without any algorithm stop criterion, having the biased as heuristic of trajectory planning and totalling 1253 mm of trajectory per layer) and the top surface of the part after printing. This geometry (a nonconvex polygon) mimics the one used by Ding et al. (2016), which is not easily printed by the traditional contour or Zigzag strategies. Six layers were printed on the top of each other with the trajectory shown in Figure 2.23(a), with a dwell time of 120 s between layers (one start and one stop per layer). The optimized trajectory basically followed a Zigzag pattern at the first half of the layer area, but with self-adjustment of the infill pattern accordingly to the distances between beads and between the end of the track and the polygon edges. The Zigzag took a slightly different pattern in the second half of the layer area, all decided by the algorithm after the 50 heuristic iterations. This was also noted at the last track, where the adjustment was made with an oscillation-like trajectory. This changing behaviour happens because the dimensions of the layers will not be always an integer multiple of the distance between beads.

Figure 2.24, in turn, presents the second geometry, based on a similar shape presented in Wang et al. (2019). This geometry is also a nonconvex polygon, yet even more complex than geometry 1, due to two inner holes. Most of the strategies presented in the introduction suggested that this geometry could not be performed by continuous printing. However, the continuous trajectory planned by using the Basic-Pixel strategy for geometry 2 was reached by utilizing 1072 nodes after 50 iterations with the four heuristics without algorithm stop criterion, having the random contour as a heuristic of trajectory planning and totalling 3038.70 mm of travel per layer. Six layers were printed on top of each other with the trajectory shown in Figure 2.24(a), with a dwell time of 120 s between layers (also just one start and stop per layer). The trajectory assumed a maze-like behaviour, in which an infill pattern made the deposition contour the circumference of the holes. It is noted that the trajectory first skirted the edges of the layer and then filled in the part, due to the fact that its origin was the Random Contour heuristic.

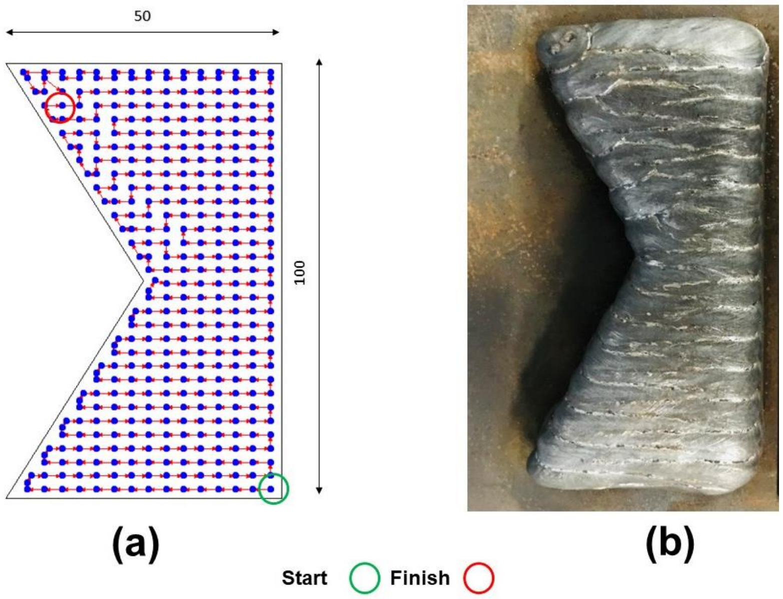


Figure 2.23 - Top view cross-section of a nonconvex polygon used as the first geometry printed using trajectory elaborated with Basic-Pixel strategy: (a) trajectory generated by Basic-Pixel strategy; and (b) 6-layer-high printed part

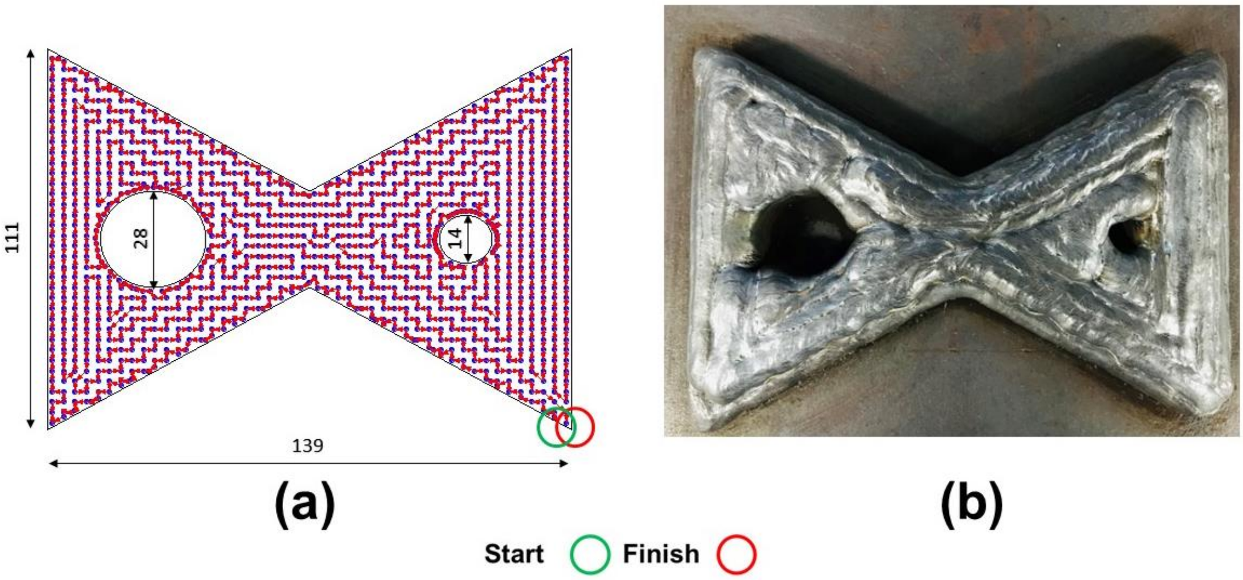


Figure 2.24 - Part with a cross-section of nonconvex polygon and holes: (a) trajectory generated by Basic-Pixel strategy; and (b) printed part by Basic-Pixel strategy

For both geometries, the Basic-Pixel strategy generated a continuous trajectory. It is noteworthy that the trajectory chosen within the matrix of best values was that with the shortest path. However, better results could have been obtained from other runs of the algorithm. This explains that the algorithm generates more than one viable trajectory for printing, in the same execution (within the array of best values) or in different executions. The parts shown in these case studies are relatively small (but complex and bulky). In this study, the maximum number of nodes was 1072, therefore, it is also important to state that a hybrid approach between Basic-Pixel and polygonal division strategy can increase the trajectory efficiency in the case of larger parts. With this hybrid approach, computational time can be reduced because the layer to be printed would be divided into smaller polygons that would have reduced numbers of nodes, which make it easier to find an optimized trajectory. This ideia will be introduced and refined in Chapter V.

Regarding the quality of the parts, Figure 2.25 illustrates the comparison between the printed parts and the 3D model, as well as the cutting surface of both parts studied. For the first criterion, the top view surfaces of the prints in Figs. 2.25(a) and 2.25(b) have in general negative deviations at the edges, possibly due to a lack of parameter optimisation for bead formation and by their uneven heating at the edges (heating dissipates more towards the inside), but this is common in parts print by WAAM. An additional layer could be deposited to compensate for this difference when the geometrical and dimensional tolerance is reached by further machining. A negative deviation can also be observed in the arc extinguishment region, particularly near the sharper corner.

It is important to mention that the largest remaining surfaces of Figs. 2.25(a) and 2.25(b) fit the target dimension (3D model outline), as highlighted in green. The green colour mesh means that the linear deviations from the external model dimensions are between ± 0.5 mm, a reasonable manufacturing tolerance. Notwithstanding, these figures also show that a more adequate parametrisation, or a correction parametrisation in offset value to generate the trajectories or mixing of strategies (for instance, Parallel Contour + Basic-Pixel hybrid strategy), is needed to make up for the low shape fitting at the acute angles. Another approach to improving the quality is to implement a closed-loop trajectory that can avoid non-conformities in the arc striking and extinguishment regions, as suggested by Hu et al. (2019). Additionally, utilizing the flexibility of the Basic-Pixel strategy to generate multiple applicable trajectories to alternate between layers, as proposed by Wang et al. (2019), this can also enhance the quality of the printed part.

In terms of the second criterion, Figures 2.25(c) and 2.25(d) illustrate the presence of a lack of fusion (highlighted in red) in some areas of the printed parts, particularly in the region between the substrate and the part. One possible explanation is that the energy input to initiate the first layer was insufficient to promote optimal bead penetration. One likely cause of the lack of fusion in other areas can be attributed to the phenomenon reported by Cui et al. (2021), where the CMT process may result in a lack of fusion in certain regions of bulk parts due to lower energy input.

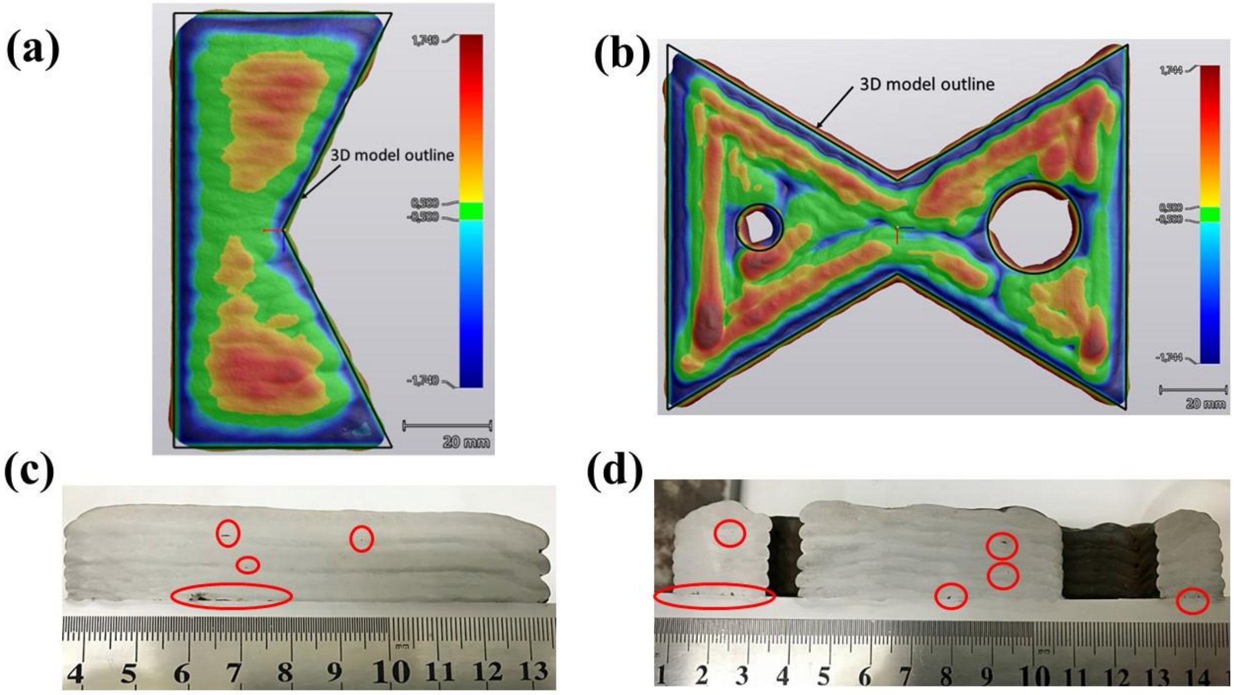


Figure 2.25 - Quality assessment of printed parts: (a) and (b) digitalized top surface of the parts showing the geometric deviations from the model outlines, and (c) and (d) longitudinal cross-section surface of the parts with emphasis on the lack of fusion

Overall, the Basic-Pixel strategy demonstrated efficacy for both geometries, but there is still room for improvement to achieve a more efficient and effective process resulting in higher-quality parts.

2.6 Partial conclusions

This chapter aimed to develop a new Space-Filling strategy for WAAM that could offer greater flexibility in trajectory planning for complex geometries using trajectory optimization algorithms, and to evaluate its effectiveness. The Basic-Pixel strategy showed to offer flexibility by providing multiple applicable trajectories that can be used to print a part. The computational evaluation showed that having four heuristics for node connections was the right decision, as the best heuristic was dependent on the geometry of the part, and all could be tested in the optimized outcome.

Regarding effectiveness, the experimental WAA-manufactured build using GMAW and plain carbon steel demonstrated that the Basic-Pixel strategy enabled the continuous deposition and construction of complex shapes, such as polygonal nonconvex geometries with holes, while maintaining surfaces typical of WAAM. The evaluation of the quality (shape, dimensional and soundness) of the parts was satisfactory, but improvements are possible, as discussed in section 2.5.2. Therefore, the next chapter will present modifications to the Basic-Pixel algorithm and printing method to enhance the quality of the parts.

CHAPTER III

ENHANCED-PIXEL STRATEGY FOR WIRE ARC ADDITIVE MANUFACTURING TRAJECTORY PLANNING: OPERATIONAL EFFICIENCY AND EFFECTIVENESS ANALYSES

3.1 Introduction, Scientific Questions, and Specific Objectives

Chapter II presented the development of a strategy named Pixel, which can be categorised as a Space-Filling strategy. In the Pixel strategy, in summary, the layer to be filled is discretised at normally equidistant nodes that will be interconnected, thus generating a continuous trajectory. In this proposal, an adapted metaheuristic Greedy Randomized Adaptive Search Procedure (GRASP) is used to find an acceptable trajectory for the deposition of the material. The metaheuristic begins with a random selection of a node (among those discretised in the layer) that will be the starting node for four competing and distinct heuristics (three of them developed by the authors for additive manufacturing). Each of the heuristics generates a trajectory, totalling four trajectories generated. Each of these generated trajectories is improved with an algorithm called 2-opt, aiming to eliminate crossings between paths and thus obtain shorter courses. After that, the trajectory with the shortest distance is taken as the best result. These steps are repeated according to the number of user-defined iterations. In the end, a list of the best results is obtained from each iteration.

The Pixel strategy is a Space-Filling strategy that can effectively fill two-dimensional layers resulting from the slicing step. One of its main advantages over other non-space-filling-driven strategies is its flexibility, which allows for modifications to be made to achieve better performance. As discussed in the previous chapter, a closed-loop strategy can be implemented to avoid issues in the arc striking and extinguishment regions. Additionally, changing trajectories between layers can also improve the quality of the printed part. However, despite these potential improvements, it is important to compare the efficiency and efficacy of the Pixel strategy with other strategies to determine its performance relative to alternatives.

Given the above, a first scientific question (SQ) arises with its respective hypothesis (H):

- SQ 3.1 - How can the Greedy Randomized Adaptive Search Procedure (GRASP) be modified to generate more diverse trajectories in the Pixel strategy and improve its performance?
- H 3.1 - By introducing diversity-promoting mechanisms such as a new ordering-axis method and heuristics of trajectory planning, the modified GRASP algorithm has the potential to generate more diverse trajectories in the Pixel strategy. This could lead to improvements in performance, such as increased quality of response.
- SQ 3.2 - How would Enhanced-Pixel (a proposed improved Pixel strategy) perform in terms of operational efficiency and effectiveness compared to conventional strategies, such as Zigzag and Parallel Contour?
- H 3.2 - The performance assessment of the Enhanced-Pixel strategy can be compared to that of Zigzag and Parallel Contour strategies using quantitative metrics, such as geometrical uncertainties, top surface waviness, and printing time.

Considering the scientific question and hypothesis mentioned, the following specific objective (SO) were proposed:

- SO 3.1 - The objective is to propose and evaluate improvements in the algorithm that may reflect on the performance gain of the original Pixel strategy;
- SO 3.2 - The objective is to investigate the operational efficiency and effectiveness of the proposed Enhanced-Pixel strategy compared to conventional strategies.

3.2 Improving the Basic-pixel strategy

To describe the accomplishment of the first specific objective of this chapter (SO 3.1), two subsections were included, namely, " Proposal to improve the Basic-Pixel strategy " and "Validation of proposed improvements", which give details of the enhancement process of the Basic-Pixel strategy.

3.2.1 Proposal to improve the Basic-Pixel strategy

The improvements proposed for the Basic-Pixel strategy were achieved through four modifications, namely, the use of the 2-opt closed-loop algorithm, the ordering direction of the nodes taking the y-axis as preferential, the introduction of a complementary trajectory planning heuristic and a choice of the heuristics' starting node.

3.2.1.1 Use of the 2-opt closed-loop algorithm

One non-conformity, not always mentioned in the Wire Arc Additive Manufacturing literature, is caused by arc striking and extinguishing. Excessive volume of material is prone in regions where the arc starts. On the contrary, a material shortage occurs where the arc is stopped. To solve this drawback, some authors have proposed solutions. For example, Zhang et al. (2003) correlated a greater bead width at the arc start region to low penetration with the GMAW process. For this reason, they proposed that the current and welding speed should be increased at the beginning of the pass. Also, according to these authors, welding current and speed must be reduced to deposit more material at the end of the layer, thus compensating for the potential lack of material. Venturini et al. (2018) proposed that arc starts and stops should be done outside the useful part, to avoid non-conformities; so, the excess/shortage can be afterwards removed by machining.

The solutions for arc starts and stops presented by Xiong et al. (2016) and Hu et al. (2018) are similar, being directed to both close and open paths. For open paths (where the multiple arc striking and extinguishing do not coincide in terms of position in the same layer), an opposite-direction movement must be adopted to deposit the following layer. Therefore, the arc strikes over the same point where the arc stopped in the previous layer, compensating for potentially generated non-conformities. In closed paths, on the other hand, the position in which the arc starts and stops already coincides. The idea of overlapping becomes interesting, as the compensation of an excess of material at the beginning of the deposition can be done by reducing the deposition of material at the end of the trajectory in the same region of the same layer.

However, these techniques become difficult for more complex geometries, generally requiring beads side by side to perform the building. This can be seen in the example illustrated in Figure 2.1, where the arc striking and extinguishing points (blue and red circles, respectively) in the same layer are never coincident. This is noticed mainly in bulky parts built up by the Zigzag and Parallel Contour strategies. It could even be argued that the mentioned drawback could be solved by building the next layer, which would follow a path opposite to that deposited in the previous layer. However, for more complex shapes, such as with holes, inversion of the deposition direction where paths without deposition will occur is not simple to implement computationally. Another argument would be in terms of productivity and print quality. For example, Jorge et al. (2022) state that when starting a subsequent layer, it is necessary to wait for the interlayer temperature to become stable to guarantee the bead geometry. But since that point with undersupply of material is the last deposited, it will probably be hotter than the other regions of the part, which can lead to a longer temperature stabilisation time (at that point), thus increasing the dwell time between layers.

To solve the problem mentioned above, the solution proposed in this work is to implement a closed-loop approach in the Basic-Pixel strategy. In the Basic-Pixel strategy, a 2-opt algorithm is already used, yet acting open-loop, to improve the initial solutions proposed by the Nearest Neighbour (NNH), Alternate (AH), Biased (BH) and Random Contour (RCH) heuristics. However, the open-cycle perspective does not form a Hamiltonian cycle (path where the start point coincides with the endpoint). Consequently, the 2-opt algorithm would find the shortest path, but the start point was not coinciding with the trajectory's endpoint. Then, using a Hamiltonian cycle seemed to be a good solution. The 2-opt algorithm seeks a shorter trajectory where the endpoint coincides with the start point (imposed constraint), as described by the classic Travelling Salesman Problem (TSP). In the next layer, another start point (which is not the same as the previous layer) can be adopted to start the deposition at a point with a more stabilised interlayer/intertrack temperature. Because of this good performance, the version Enhanced-Pixel strategy adopted this solution.

3.2.1.2 Ordering direction of nodes taking the y-axis as preferential

By default, the Basic-Pixel strategy preferentially orders the interconnectable nodes generated on the layer surface towards the x-axis (row-oriented). Figure 3.1(a) demonstrates the sorting process along the x-axis, where $\boldsymbol{i_1}$ and $\boldsymbol{i_n}$ are the first and last nodes, respectively, of the first row of nodes (from bottom to top). The node $\boldsymbol{i_{n+1}}$ represents the first one of the second row of nodes. The algorithm organises such nodes in rows, indexing them to the values of y. The nodes are sorted in ascending order from left to right in each row.

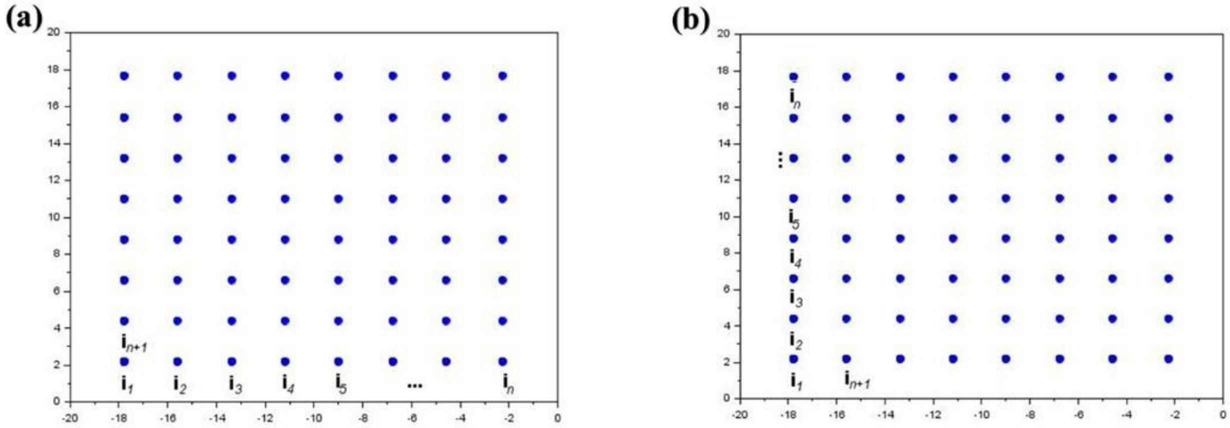


Figure 3.1 - Ordering of discrete nodes on the layer: (a) towards the x-axis (used in Basic-Pixel); and (b) towards the y-axis (new ordering proposal employed in the Enhanced-Pixel)

However, a hypothesis raised is that other forms of axis ordering methods could favour the performance of the Basic-Pixel strategy, that is, generate shorter trajectories. An additional arrangement that uses the y as a preferential axis (columns oriented) is proposed in this context. In this case, the interconnectable nodes are organised in columns indexed to the x-values. For each column, there is a number of discrete nodes, which are sorted from bottom to top. As a preferential column, sorting starts on the column with the lowest x-value and moves to the column with the highest x-value (moving from left to right). Figure 3.1(b) demonstrates the sorting process in the y-axis direction, where ***i_1*** and ***i_n*** are the first and last nodes, respectively, of the first column of nodes (from left to right). The node ***i_{n+1}*** represents the first one of the second column of nodes.

With these two extra options for ordering the nodes, it is believed that a performance improvement of the Basic-Pixel strategy can emerge, allowing further exploration of the response space during optimisation.

3.2.1.3 Introduction of a complementary trajectory planning heuristic

The Basic-Pixel strategy has four heuristics, namely NNH (Nearest Neighbour), AH (Alternating), BH (Biased) and RCH (Random Contour). These heuristics are applied to plan trajectories before going through the optimisation stage, through the 2-opt algorithm. The better the initial solution, the easier the local search efficiency (e.g., by 2-opt). Therefore, a new heuristic, Continuous (CH), is proposed to boost the Basic-Pixel strategy.

Figure 3.2 shows the flowchart of how this planned heuristic works in the Enhanced-Pixel strategy. The process begins with selecting an initial node (the starting node of the trajectory), which will be considered the active node. Once the active node is defined, the heuristic will select the closest node among all the nodes not yet visited (not interconnected). With this selection performed, the active node links to the selected node forming a path, and this last node becomes the active node for the beginning of the process again, until all nodes are linked. In the case of a tie in the selection step between unvisited nodes closest to the active node, the unvisited node with the highest sorting index will be chosen.

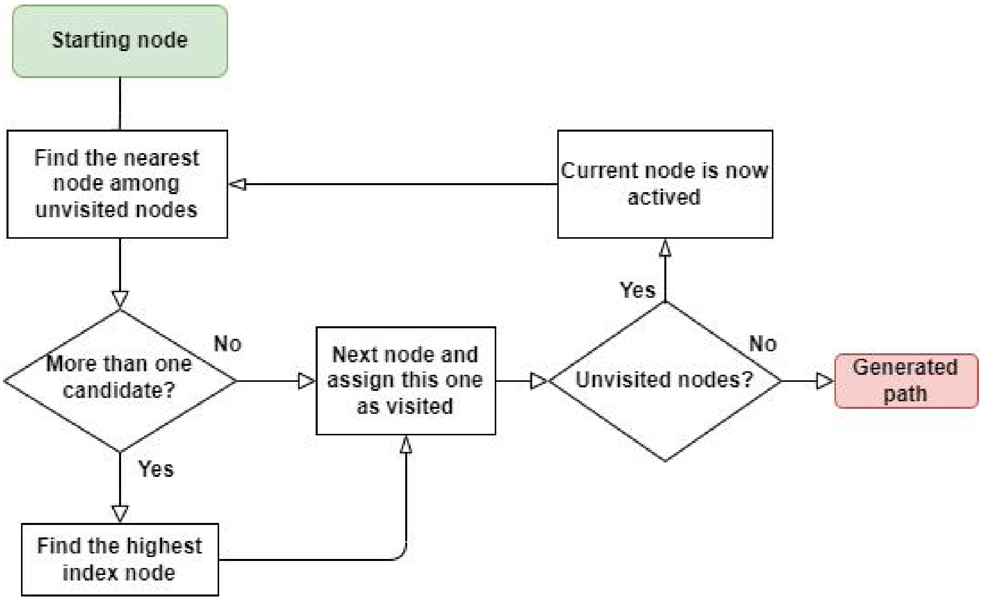

Starting node
Find the nearest node among unvisited nodes
Current node is now actived
Yes
No
More than one candidate?
Next node and assign this one as visited
Unvisited nodes?
No
Generated path
Yes
Find the highest index node

Figure 3.2 - Flowchart of the continuous trajectory planning heuristic (CH)

3.2.1.4 Choice of the heuristics' starting node

The self-selection of the initial node, which is the position where the heuristics of trajectory planning will start the trajectory generation, was done randomly in the Basic-Pixel strategy. However, a relevant question would be whether any specific position of the initial node (starting node) could influence the performance of the Basic-Pixel strategy in such a way as to generate shorter trajectories. Therefore, following this work systematism, it was also implemented, and assessed, an optional procedure for the starting node for heuristics in the Enhanced version of Pixel to verify whether the initial node specific positioning is relevant.

3.2.2 Validation of the proposed improvements

3.2.2.1 Validation procedure

Figure 3.3 schematises the workflow of the novel algorithm proposed for the Enhanced-Pixel strategy. As seen, after randomly choosing the start node, 10 combinations of node ordering direction (row-oriented and the included column-oriented) and 5 heuristics of trajectory planning (including the proposed Continuous (CH)), are performed before applying the closed-loop 2-opt algorithm optimisation. The output is the shortest distance stored in the "Best value matrix", sorted out those achieved in "store values in a matrix". It is important to note that two checking points are provisorily planned in the workflow, represented in the figure by cylinders (highlighted with dashed lines) on the right side. The first one is the "Node Classification", which indicates the nodes drawn at each iteration to be the starting nodes of the heuristics. They are characterised as 0, for nodes belonging to the contour of the set of nodes, or 1, for the other nodes. The second checking point is referred to as "Trajectory Distance" after each combination of iteration and heuristic. The information from each provisory checking point made it possible to perform a statistical analysis relating the starting nodes' location and the trajectory's distance. Thus, it is possible to study if there is any preferred position for the beginning of the trajectory in its initial phase.

Three printable parts (Figure 3.4) were studied to assess the feasibility of the Enhanced-Pixel strategy (with all improvement proposals implemented to the Basic-Pixel strategy). Note the differences in the designs of each part, highlighting a variety of complexities for the printing process (consequently, the trajectory planning).

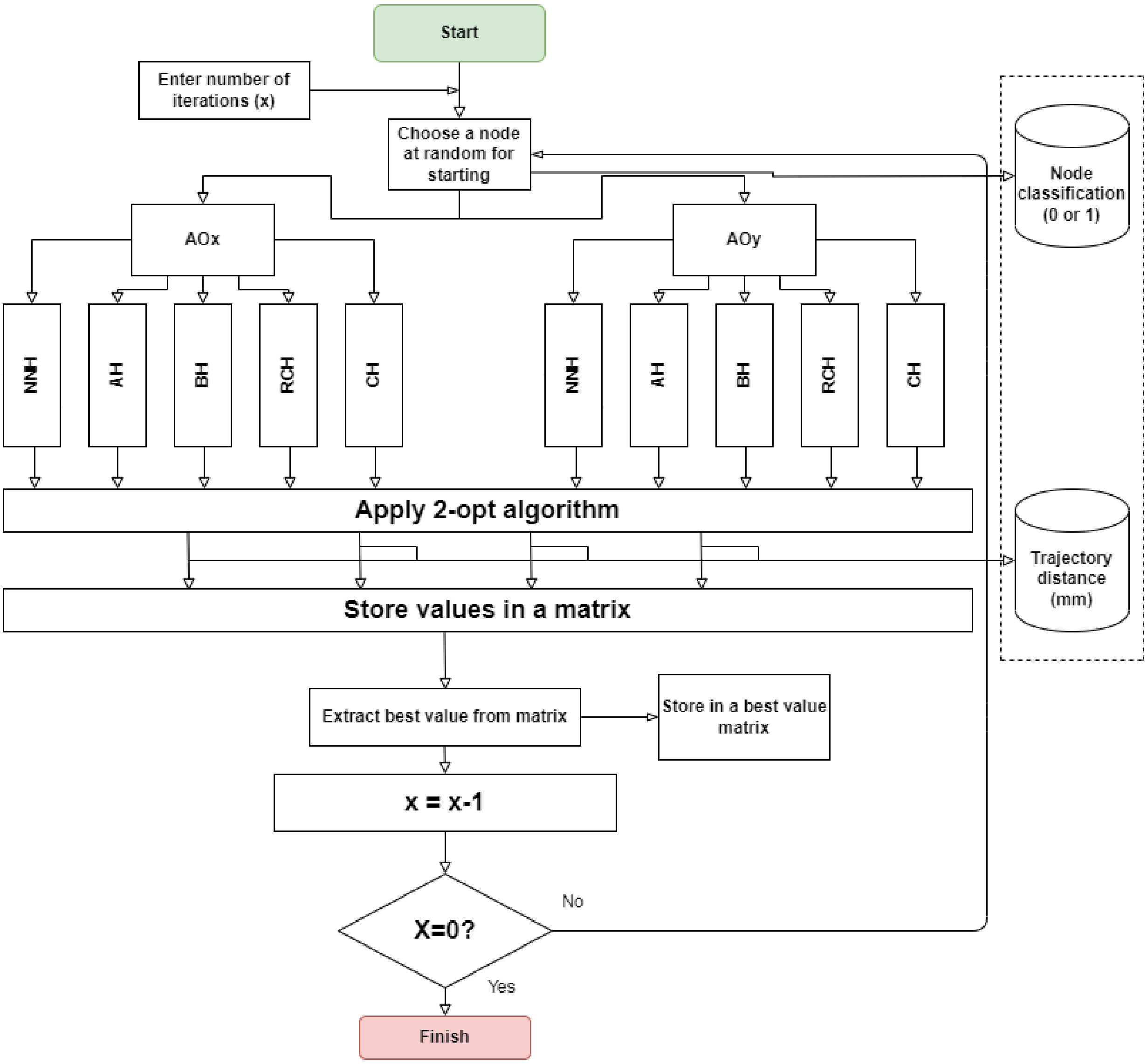

Start
Enter number of iterations (x)
Choose a node at random for starting
AOx
AOy
NNH
AH
BH
RCH
CH
NNH
AH
BH
RCH
CH
Apply 2-opt algorithm
Store values in a matrix
Extract best value from matrix
Store in a best value matrix
x = x-1
X=0?
No
Yes
Finish
Node classification (0 or 1)
Trajectory distance (mm)

Figure 3.3 - Flowchart of the GRASP metaheuristic representing the Enhanced-Pixel strategy, with the proposed improvements implemented to the Basic-Pixel (where, NNH – Nearest Neighbour heuristic; AH – Alternate heuristic; BH – Biased heuristic; RCH – Random Contour heuristic; CH – Continuous heuristic heuristics; AO – Axis ordering, either in axis x or in axis y)

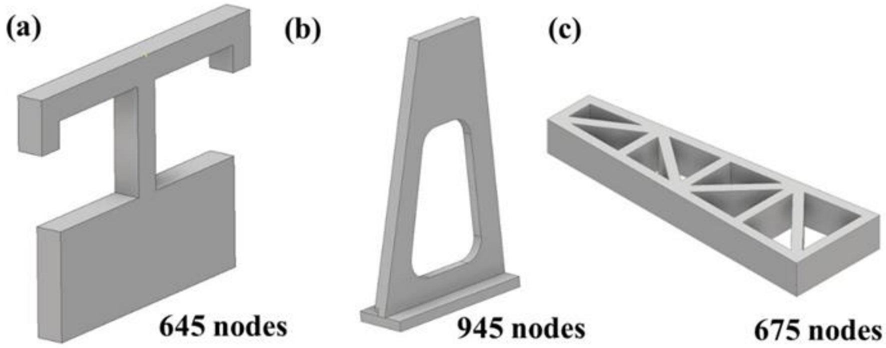
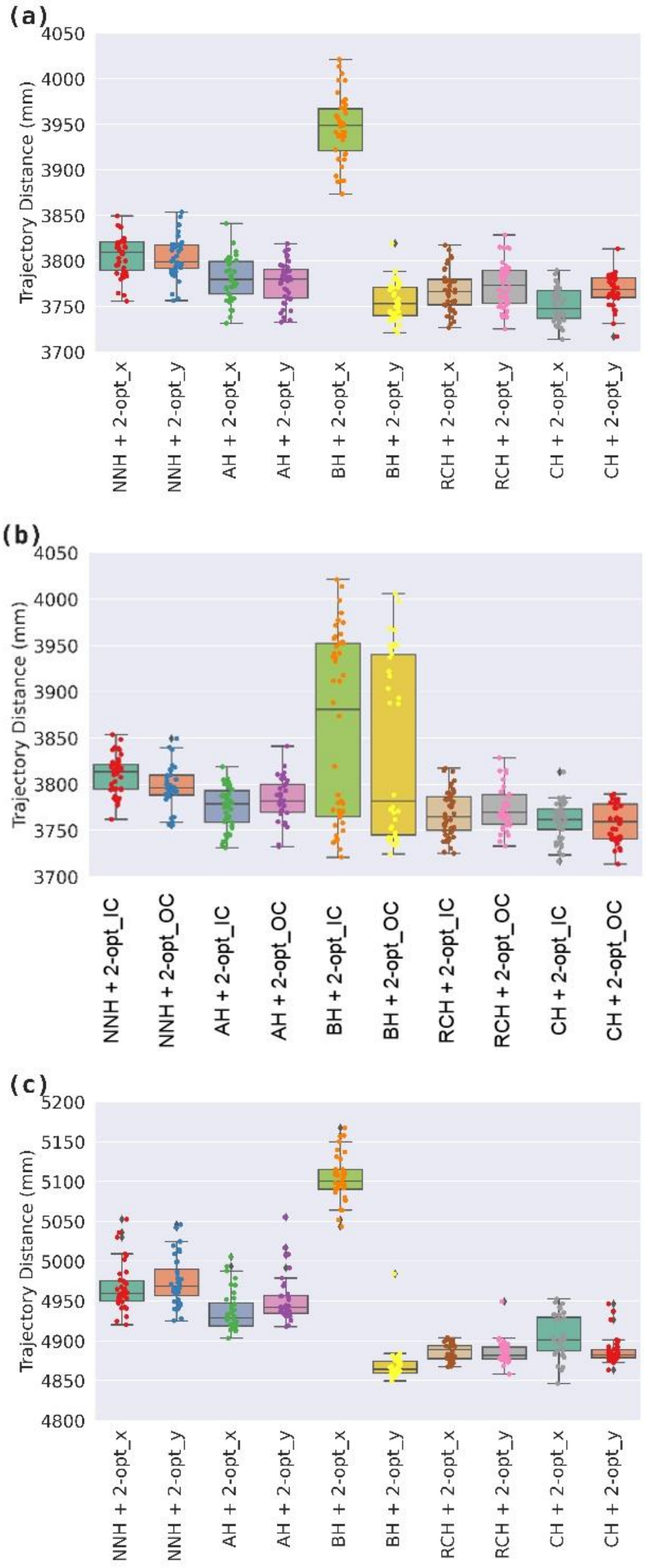


Figure 3.4 - Selected printable parts with different complexities to validate the Enhanced-Pixel strategy: (a) "Part 1", with narrow and bulky regions; (b) "Part 2", not symmetrical with hole; and (c) "Part 3", in the form of a truss and composed of thin walls (remembering, the number of nodes depends on the spacing and layer area)

3.2.2.2 Validation Results and Discussions

Forty iterations were arbitrarily defined to perform the trajectory simulation of each part, according to Figures 3.3 and 3.4. Figure 3.5 shows the boxplots (a standardised way of displaying data distribution based on the minimum, first quartile, median, third quartile, and maximum) of the resulting trajectory distances (in mm). Figures 3.5(a), (c) and (e) relate the trajectory distance with the heuristic used for a sorted axis. The symbology to indicate each heuristic used are coded as LL+N-xxx_e, where "LL" means the type of heuristic (NNH for Nearest Neighbour, AH for Alternate, BH for Biased and RCH for Random Contour), "N -xxx" that was optimised by the 2-opt algorithm and "_e" that the nodes were ordered in relation to the x or y-axis (_x or _y, respectively) or that the trajectory generation node started at a position on the contour (_IC) of the layer or started outside (inward) the contour (_OC). For example, the name AH+2-opt_x, indicates that the Alternate heuristic (AH) was used and the nodes were ordered in relation to the x-axis and optimised by the 2-opt algorithm. Figures 3.5(b), (d) and (f), in turn, relate the trajectory distance with the heuristic used for a given position of the starting node. For example, the acronym BH+2-opt_IC, indicates that the trajectory generation node started on the contour (IC) of the layer and the Biased heuristic (BH) was used to generate the trajectory, later optimised by the 2-opt algorithm. The opposite is true for the denomination BH+2-opt_OC, where the trajectory generation node started outside the contour (OC).


(a)
Trajectory Distance (mm)
4050
4000
3950
3900
3850
3800
3750
3700
NNH + 2-opt_x
NNH + 2-opt_y
AH + 2-opt_x
AH + 2-opt_y
BH + 2-opt_x
BH + 2-opt_y
RCH + 2-opt_x
RCH + 2-opt_y
CH + 2-opt_x
CH + 2-opt_y
(b)
Trajectory Distance (mm)
4050
4000
3950
3900
3850
3800
3750
3700
NNH + 2-opt_IC
NNH + 2-opt_OC
AH + 2-opt_IC
AH + 2-opt_OC
BH + 2-opt_IC
BH + 2-opt_OC
RCH + 2-opt_IC
RCH + 2-opt_OC
CH + 2-opt_IC
CH + 2-opt_OC
(c)
Trajectory Distance (mm)
5200
5150
5100
5050
5000
4950
4900
4850
4800
NNH + 2-opt_x
NNH + 2-opt_y
AH + 2-opt_x
AH + 2-opt_y
BH + 2-opt_x
BH + 2-opt_y
RCH + 2-opt_x
RCH + 2-opt_y
CH + 2-opt_x
CH + 2-opt_y

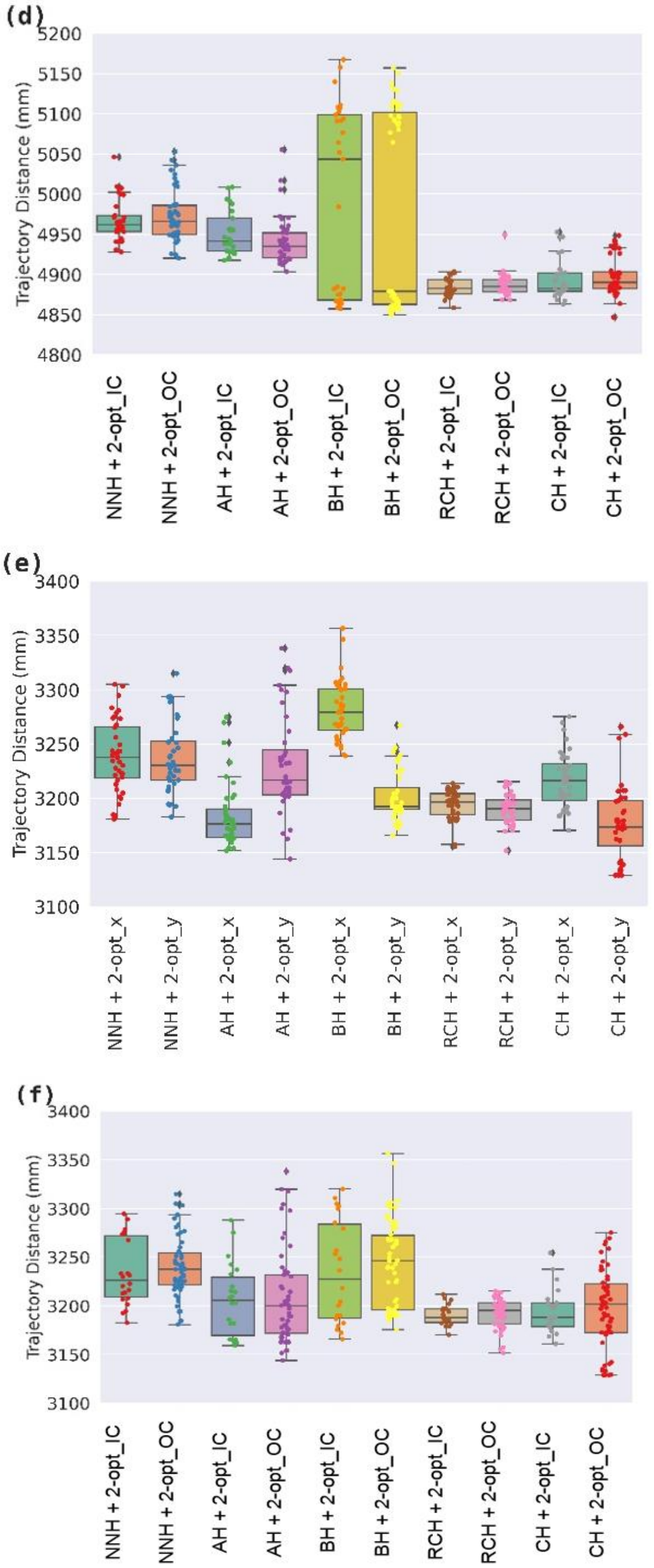


Figure 3.5 - Boxplots of path distances obtained by the heuristics by axis ordering (plots on the left) and by starting node positioning (plots on the right), where: (a) and (b) refer to "Part 1"; (c) and (d) to "Part 2"; and (e) and (f) to "Part 3"

The median (central line of the coloured box) was the only statistics used for the data analysis as a comparison parameter. In "Part 1", the ordering of the axes significantly influences the distance of the trajectories in the Biased (BH) and Continuous (CH) heuristics, as can be seen in Figure 3.5(a). For the other heuristics, the median results are very similar. The best result found was with the Continuous heuristic and the nodes ordered on the x-axis. The worst result found was with the Biased heuristic and the x-axis node ordering. Regarding the analysis of the position of the initial node, no significant differences were found (between the heuristics) in relation to the position of the initially chosen node, as seen in Figure 3.5(b). However, as will be seen later, the performance of each heuristic depends on the part geometry.

Regarding "Part 2", one can be seen in Figure 3.5(c) that the order of the axes significantly influenced the performance of the Enhanced-Pixel strategy when employing the Biased (BH) and Continuous (CH) heuristics. For the other heuristics, the median results are very similar. The best result found for "Part 2" (at the given number of iterations) was with the Continuous heuristic with the nodes ordered on the x-axis. The worst result found was the Biased heuristic with an x-axis ordering. In the initial node's position analysis, according to Figure 3.5(d), no significant differences were found (between the heuristics) of the position of the node initially chosen.

Figure 3.5(e), in turn, indicates that for processing "Part 3" that the ordering of the axes presents significant differences in the Alternating (AH), Biased (BH) and Continuous (CH) heuristics. For the other heuristics, the median results are very similar. The best result found was with the Continuous heuristic with the nodes ordered on the y-axis. The worst result found was the Biased heuristic with the ordering in relation to the x-axis. In the analysis of the initial node position for "Part 3", according to Figure 3.5(f), no significant differences (between the heuristics) in the position of the node initially chosen were also found.

In view of the above, it is noted that for all three different parts shapes under study, the Nearest Neighbour heuristic (NNH) has a median that is further away from the others; their minimum values can still be considered acceptable, as they are below the median of other heuristics-sorted axis combinations. Notwithstanding, there is no evidence that this outcome would repeat with a different part geometry and/or a higher number of iterations. This reinforces the need for the presence of all heuristics, including the Continuous one introduced in this work. It can also be inferred that both axis ordering methods (x and y) can generate satisfactory results, depending on the layer geometry type. However, regarding the location of the initial node (belonging or not to the contour), the case study indicated that it does not influence the performance of the Enhanced-Pixel strategy, and both methods can be used (for safety's sake, better to keep this option).

Therefore, the Enhanced-Pixel strategy benefited from all the improvements proposed in section 3.2.1.

With this basic visual analysis done by boxplot, it can be inferred that the insertion of the Continuous (CH) heuristic and the y-axis ordering can increase the performance of the Basic Pixel strategy. However, it is worth mentioning that more advanced statistical analyses would not be feasible for this study, since the objective is not to say which is the best combination between axis ordering and heuristics of trajectory planning. One must remember that the number of nodes and the part topology may lead to either one or the other combinations as the better (as seen in the results). That said, a more straightforward method for the best combination selection between axis ordering and heuristics of trajectory planning seemed to be that one of the optimisation processing.

3.3 Operational efficiency

To partially address the second specific objective of this chapter (SO 3.2), this section focuses on defining the WAAM processing qualification (assessment criterion) based on the operational efficiency of trajectory planning, specifically in terms of trajectory distance, building time, and non-deposition paths. The efficiency of trajectory planning is determined by the torch's shorter movement distance, faster speed, and minimal non-deposition paths, which are crucial for optimizing WAAM processing. Note that this assessment criterion does not evaluate the quality of printed parts (the focus of section 3.4) but rather the efficiency of trajectory planning.

3.3.1 Methodology

The three parts presented in Figure 3.4 were used again as case-study to assess the operational efficiency of the trajectory planning programme. In addition to Enhanced-Pixel, the Parallel Contour and Zigzag strategies were used to generate simulated trajectories, using an upgraded of the in-house software described in Ferreira et al. (2022). Particularities of the trajectory generation were adopted for all printable parts. Using the Zigzag strategy, the scanning angles of 0° and 90° (in relation to the x-axis) were alternately used in the deposition of the layers. With the strategy of Parallel Contour, the inside-out and outside-in build approaches were used alternately in the deposition of the layers (naturally, they have the same trajectory, inverting only the sense of the movement). And finally, in the Enhanced-Pixel strategy, the two best trajectories were selected to alternate during layer deposition, one applied to the odder layers and the second to the even layer. It is noteworthy that arbitrarily only two alternating trajectories between layers were employed, regardless of the strategy.

The input parameters to run the programme, which include the stepover distance (distance between two consecutive bead centres), are presented in Table 3.1. Forty iterations were applied. The operational efficiency criterion were evaluated through offline simulations of the building process performed in the SprutCAM software.

Table 3.1 - Input parameter for offline simulation of case-study parts

Input parameter	"Part 1"	"Part 2"	"Part 3"
Bead height (mm)	2.49	2.30	2.42
Bead width (mm)	6.60	7.45	5.55
Deposition speed (cm/min)	40	40	40
Stepover distance (mm)	4.87	5.5	4.10

3.3.2 Results and Discussions

Figure 3.6 shows the result of the offline simulations with their respective operational efficiency parameters for the strategies applied to generate trajectories for "Part 1". It is noted that the Enhanced-Pixel strategy presented the shortest trajectory distances, but it does not mean the shortest building time. To explain this apparent contradiction, Enhanced-Pixel produced a greater number of direction changes than the Zigzag strategy, in which the torch performs decelerations and accelerations that increase the building time. However, when analysing together the trajectories used to build the even and odd layers, the total building time performed with the Enhanced-Pixel strategy (Figure 3.6(d) and (e)) exceeds in 1 second the building time performed with the Zigzag strategy (Figure 3.6) (a) and (b)), i.e., 1000 s against 1001 s (see Table 3.2). It is important to mention that this building time difference is not significant. The Parallel Contour strategy, in turn, presents the longest building time. Regarding the number of non-deposition paths, Parallel Contour and Enhanced-Pixel strategies surpassed the others.

When the offline simulation was applied to "Part 2", Figure 3.7 shows that the Zigzag strategy reached the lowest building time and presented one non-deposition path at each layer. The Parallel Contour strategy did not appear suitable for building this part because of the voids left inside, besides offering the longest trajectory distance and two non-depositions paths. The Enhanced-Pixel strategy generated shorter trajectories distance (Figure 3.7(e) and (f)) and with no arc interruptions, but with slightly higher building time.

Concerning "Part 3", data from Figure 3.8 suggests that the Zigzag strategy obtained the shortest trajectory distance (Figure 3.8(b)) and the least building time, but presented several non-deposition paths (above ten). The Parallel Contour strategy was not suitable for building this part due to the voids left at the intersections of the truss bars (a common problem reported in the literature), but it presented the shortest trajectory distance and building time, although it totalled 7 non-deposition paths. The Enhanced-Pixel strategy generated continuous trajectories, but with slightly higher building time.

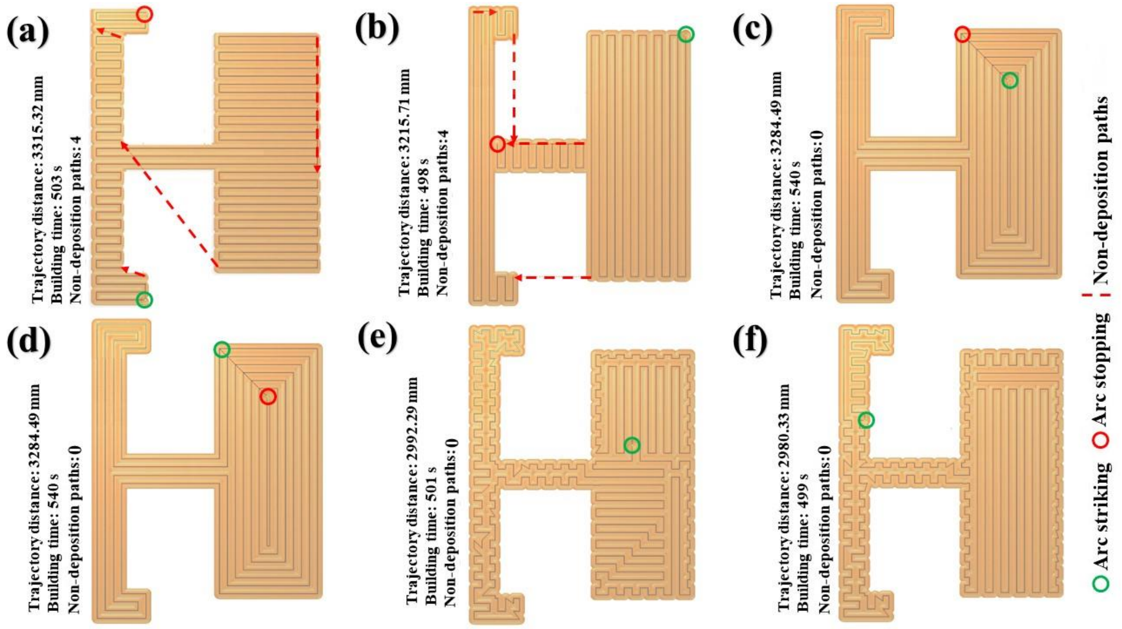


Figure 3.6 - Offline simulations of trajectories for "Part 1", using the strategies: (a) Zigzag 0°; (b) Zigzag 90°; (c) Parallel Contour inside-out; (d) Parallel Contour outside-in; (e) Enhanced-Pixel for odd layers; and (f) Enhanced-Pixel for even layers

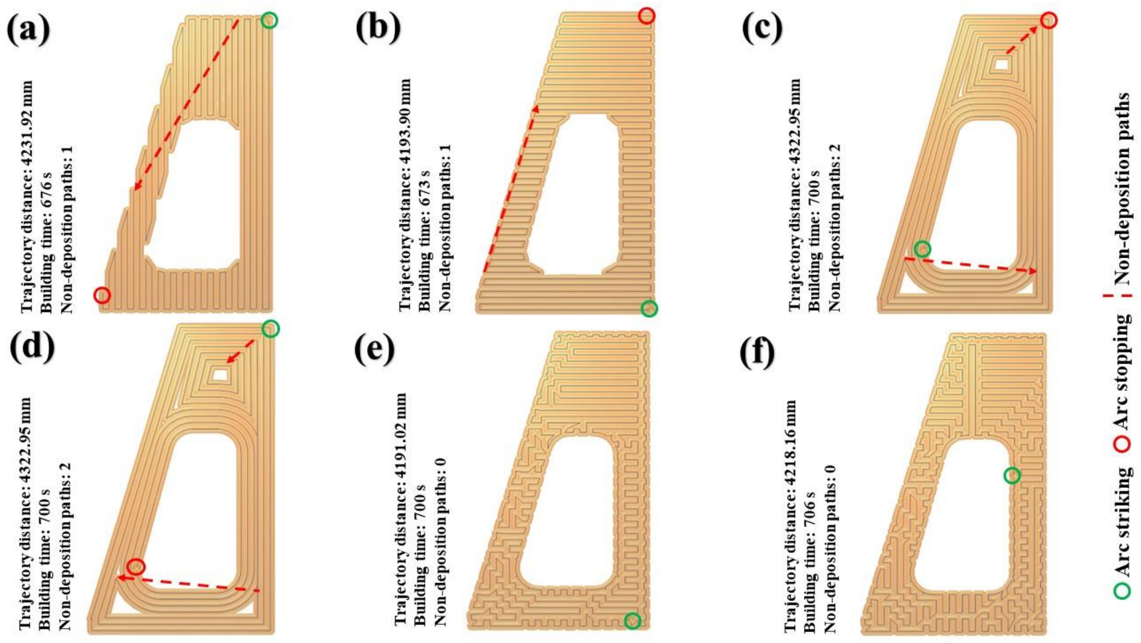


Figure 3.7 - Offline simulations of trajectories for "Part 2", using the strategies: (a) Zigzag 0°; (b) Zigzag 90°; (c) Parallel Contour inside-out; (d) Parallel Contour outside-in; (e) Enhanced-Pixel for odd layers; and (f) Enhanced-Pixel for even Layers

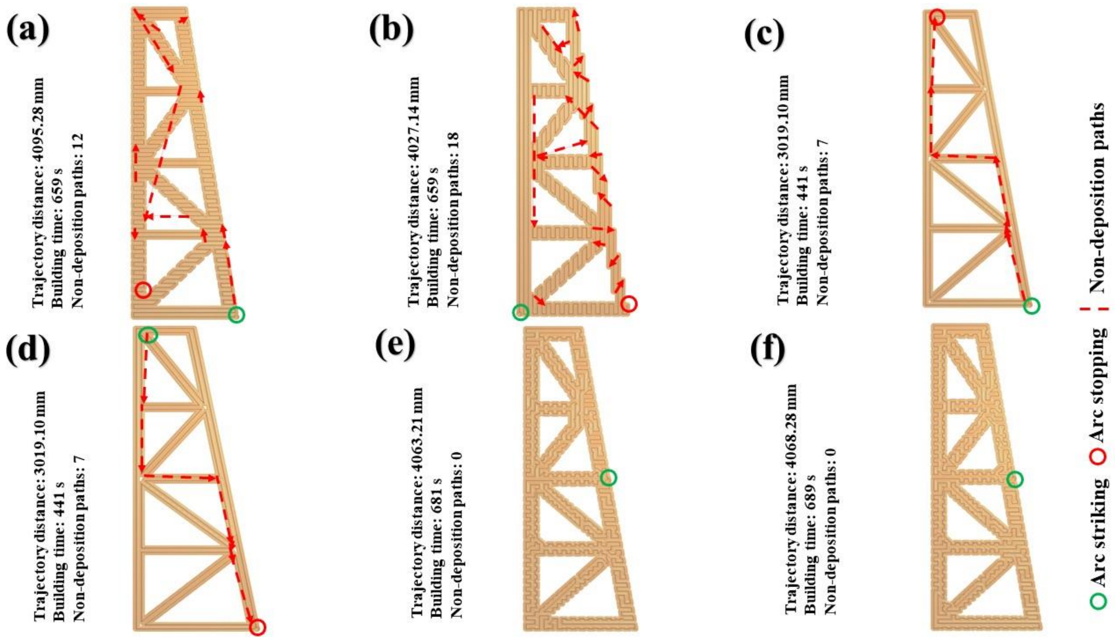


Figure 3.8 - Offline simulations of trajectories for "Part 3", using the strategies: (a) Zigzag 0°; (b) Zigzag 90°; (c) Parallel Contour inside-out; (d) Parallel Contour outside-in; (e) Enhanced-Pixel for odd layers; and (f) Enhanced-Pixel for even Layers

Gathering all printable part results together, shown in the Table 3.2, in general, the Zigzag strategy presented good performance concerning the trajectory distance and building time, superior to the other strategies when printing Parts 2 and 3. However, it should be noted that the Zigzag strategy, used in more complex parts, can leave unfilled regions (the built part does not correspond to the measurements of the 3D model). According to Xiong et al. (2019), the reason for this is the distance between the scan vectors, which may not be multiple of the part's regions to be filled with. This setback worsens when this strategy is used as a segment of the hybrid contour strategy, as shown in Figure 3.9; the unfilled regions are clearly left inside the part. One solution would be to manufacture the parts with an adequately calculated over metal. However, one should be aware that this will increase material usage and building time. Concerning non-deposition paths, non-conformities in geometry may occur in the arc striking and stopping regions, as cited by Hu et al. (2018). In the case of "Part 3" trusses, this can be an aggravating factor due to the high number of non-deposition paths (see Table 3.2).

The Parallel Contour strategy is potentially not applicable for more complex parts, not due to efficiency-related issues, but because of the proneness of imperfections. For instance, in the case of "Part 1" (Figure 3.6(c) or (d)), the start/beginning of the deposition became narrower than the offset distances calculated by this strategy between the tracks. Therefore, there will occur a material accumulation in this position, which will put at risk the entire building process. In "Part 2," the voids left by this strategy (Figure 3.7(c)) are critical. However, a possible solution for printing using the concept of Parallel Contour would be to use the A-MAT strategy presented by Ding et al. (2016). In relation to "Part 3", the voids left at the truss intersections bars can be solved with an adaptation of the Parallel Contour algorithm to make corrections at the intersections, as proposed by Nguyen et al. (2020), in addition to promoting continuous deposition. However, it is worth mentioning that all suggestions for improvements would come true from experimental tests, that is, there is a need to know the input parameters (e.g., material, process, wire diameter, deposition speed, wire feed speed) and the respective output (e.g., bead height and width) of the process. In the case of A-MAT, trajectories are generated with scan lines that are not equidistant from each other, so there is a need to change parameters during the deposition to have these spaces filled. Concerning Nguyen et al. (2020) 's solution, the displacement of trajectories in angle corners is not calculated but rather experimentally raised and predicted with the aid of machine learning algorithms.

Table 3.2 - Criteria data for all parts

Parts	Strategies	Criteria			Sum of two layers per criterio		
		Trajectory Distance (mm)	Building time (s)	Non-deposition paths	Trajectory Distance (mm)	Building time (s)	Non-deposition paths
"Part 1"	Zigzag 0°	3315.32	503	4	6531.03	1001	8
	Zigzag 90°	3215.71	498	4			
	Contour inside-out	3284.49	540	0	6568.98	1080	0
	Contour out-inside	3284.49	540	0			
	Enhanced-Pixel odd layers	2992.29	501	0	5972.62	1000	0
	Enhanced-Pixel even layers	2980.33	499	0			
"Part 2"	Zigzag 0°	4231.92	676	1	8425.82	1349	2
	Zigzag 90°	4193.90	673	1			
	Contour inside-out	4322.95	700	2	8645.90	1400	4
	Contour out-inside	4322.95	700	2			
	Enhanced-Pixel odd layers	4191.02	700	0	8409.18	1406	0
	Enhanced-Pixel even layers	4218.16	706	0			
"Part 3"	Zigzag 0°	4095.28	659	12	8122.82	1318	30
	Zigzag 90°	4027.14	659	18			
	Contour inside-out	3019.10	441	7	6038.20	882	14
	Contour out-inside	3019.10	441	7			
	Enhanced-Pixel odd layers	4063.21	681	0	8131.49	1370	0
	Enhanced-Pixel even layers	4068.28	689	0			

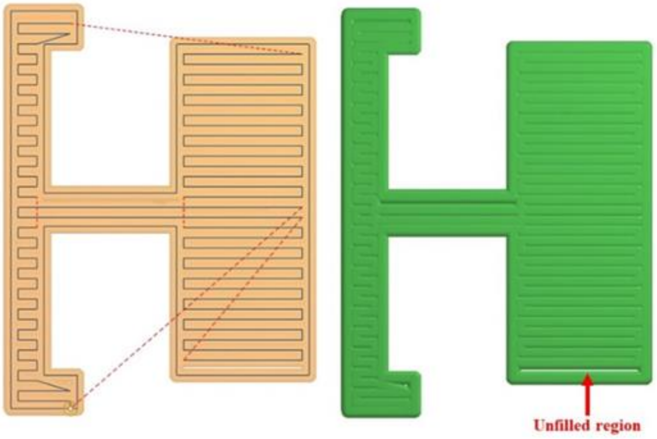


Figure 3.9 - Unfilled region in "Part 1" due to the non-multiplicity of scan vectors in the Zigzag strategy, in a hybrid Parallel Contour strategy

Finally, a good balance between the trajectory distance and building time was reached with the Enhanced-Pixel strategy. In the most critical cases, Parts 2 and 3, the difference to the Zigzag strategy did not exceed 5% of the total building time per layer, which represented in this case study less than 30 s (about 1 minute for two layers, as shown in Table 3.2). However, the Zigzag strategy promoted layer regions that were not filled by the set of scan vectors, reducing time and distance. The Enhanced-Pixel strategy has ensured a complete filling of a layer, because the algorithm discretises the entire layer with interconnected nodes to generate a trajectory. In addition, the Enhanced-Pixel strategy generates continuous trajectories for all parts studied, avoiding non-conformities from arc interruptions.

As demonstrated with the above simulations using different shapes of printable parts and trajectory planning strategies, operational efficiency analysis is complex. The criteria are not harmonious and directly related. Satisfactory attendance of a criterion with one strategy does mean that the others will be attended with the best performance. A good performance of one criterion can disguise imperfections in the built part (for example, an unfilled region due to the lack of a scan line) that might be noted only in the actual processing. Therefore, the qualification of operational effectiveness of the proposed Enhanced-Pixel strategy was conducted in a holistic manner. At least for the current studied cases, the Enhanced-Pixel strategy, compared to the Zigzag strategy, can be considered very efficient, with competitive building times and moving distances, not presenting non-deposition paths (continuous deposition). In addition, the Zigzag strategy is more prone to imperfections.

3.4 Operational effectiveness

To achieve the second specific objective of this chapter (SO 3.2), this section focuses on operational effectiveness, which can be defined as achieving outputs that meet the intended quality standards and are in accordance with the design specification. This section aims to assess the operational effectiveness of the Enhanced-Pixel strategy in geometrical uncertainties and top surface waviness of a part, compared to the Zigzag trajectory planning strategy. This evaluation is based on the premise that while Zigzag is operationally efficient (as shown in the previous section), Enhanced-Pixel may offer improved effectiveness in achieving the desired part characteristics.

3.4.1 Methodology

"Part 1" presented in Figure 3.4(a) was arbitrarily selected to be physically built (and not simulated) according to the strategies presented in the operational efficiency section, Figure 3.6(a) and (b) related to the strategy Zigzag, and Figure 3.6(d) and (e) to Enhanced-Pixel strategy. The part was not also built with the Parallel Contour strategy due to the non-conformities already created in the offline simulations.

The dimensional details of the built parts are presented in Figure 3.10. Deposition passes (tracks) 2.49-mm high and 6.60-mm wide were generated, according to the experimental configurations presented in Table 3.3. According to these dimensions and the overlap model proposed by Ding et al. (2015d), the stepover distance was configured at 4.87 mm.

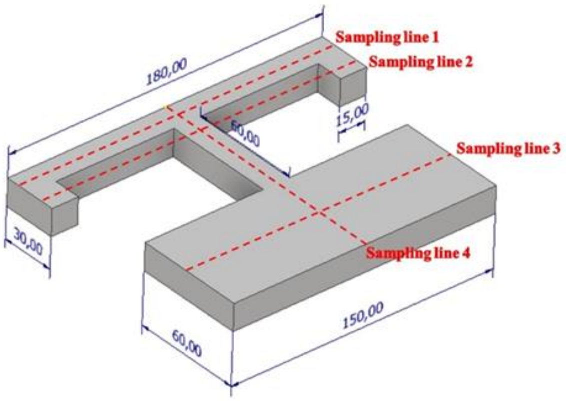


Figure 3.10 - "Part 1" with its respective dimensions (in mm) and sampling lines for dimensional analyses (top surface waviness and geometrical uncertainties)

Table 3.3 - Experiment settings for building "Part 1"

Process	Pulsed-GMAW
Arc deposition equipment	Fronius CMT – TransPuls Synergic 500
Rig	3-axes CNC gantry with a working envelope of 500 mm x 500 mm x 300 mm
Substrate	SAE 1020 carbon steel (200 mm x 200 mm x 12 mm)
Substrate cooling	Active cooling (NIAC*)
Feedstock (Wire)	AWS ER90S-B3, ϕ 1.2 mm
Shielding Gas	Ar + CO_2 (4%) at 15 L/min
CTWD**	17 mm
Deposition speed	40 cm/min
Wire feed speed	5.2 m/min
Average current	175 A
Set voltage	23 V
Interlayer Temperature	80 °C (monitored with a pyrometer along the sampling lines shown in Fig. 3.10)

(*) NIAC - near-immersion active cooling, described in Silva et al. (2020); (**) – CTWD - contact tip to work distance

In addition, Figure 3.10 shows 4 sampling lines that were arbitrarily defined for the dimensional analysis. Sampling lines 1 and 2 pass through specific regions of the T-intersection that, according to Nguyen et al. (2020), may present areas conducive to geometric non-conformities. Sampling line 3 passes exclusively through the bulky region, and sampling line 4 passes longitudinally through a part containing narrow and bulky regions. To compare the operational effectiveness between Zigzag and Enhanced-Pixel strategies, the profiles of the cross-sections from where the sampling lines spanned were initially scanned using a 3D HandyScan 3DTM scanner from Creaform.

The geometrical uncertainty metric was determined by visually comparing the outlines of the four sampling lines. The meshes obtained by the scanning were compared with the 3D model surface using the VXElements software, from the same scanner company. A program, developed in Python, was used to perform the analyses on the sampling lines. The second metric was the top surface waviness (TSW). To quantify this metric, only sampling lines 1, 3 and 4 were used (sampling line 2 did not have enough length to characterise a theoretically planar surface). A TSW parameter was calculated from the 3 scanned sampling lines. For that, the know metrology arithmetic average roughness (Ra) concept was used. In short, the average arithmetic roughness (Ra) is the arithmetic mean of the absolute values of the ordinates of the effective (measured) profile in relation to the midline in a sample length (ALMEIDA et al., 2018). Equation (3.1) transfers this concept to TSW, where n is the number of discretised samples per length of the sampling lines, H is the height of the midline between peaks and valleys, and h is the height of each sampled point (peaks and valleys) along the same line length.

$$TSW = \frac{1}{n}\sum_{i=1}^{n}|h_i - H| \tag{3.1}$$

3.4.2 Results and Discussions

Figure 3.11 shows the built parts and their respective digitised meshes, indicating the geometric deviation characterised by the relative distance between the built part and the 3D model. The part created with the Enhanced-Pixel strategy, Figure 3.11(c) and (d), presents a larger region with green colouration, which indicates that most of the build deviation from the 3D model is within a range from – 1.0 mm to + 10 mm (range chosen arbitrarily in this work). The part built by the Zigzag strategy, Figure 3.11(a), presented an unfilled region, as highlighted by a red circle in Figure 3.11(b), reaching a deviation up to -2.0 mm. This was probably due to the absence of a scan vector in that region to fill the part in the trajectory shown in Figure 3.6(a) (x-axis 0° scan vectors) and did not support the beads deposited in the next layer, using 90° scan vectors (Figure 3.6(b)). Then, a molten pool collapse probably occurred in this area. It can also be also noted in Figure 3.11(b) regions with higher deviations (in dark blue, which characterise dimensional deviations of around -- 2.0 mm). High deviations (dark blue) are also found in Figure 3.11(b), enclosed in the narrow regions. Those regions are close to the arc striking and extinguishing, previously identified in Figure 3.6(a) and (b). Hu et al. (2018) say this may cause geometric irregularities. It is noteworthy that continuously building a layer using the conventional Zigzag strategy is impossible. Recently Wang et al. (2019) presented an algorithm that can generate a zigzag with continuous scan for WAAM. Gomez et al. (2022) suggested something similar, an algorithm that can continuously generate a hybrid scan (Contour + Zigzag). However, this author assessed this approach only on polymers, yet promising for WAAM application.

The Enhanced-Pixel strategy led to a satisfactory dimensional aspect, probably due to both the trajectory variation between layers and a continuous trajectory. In agreement with the hypothesis presented, Wang et al. (2019) state that the variations in the trajectory between layers contribute to reducing the accumulation of height errors on the surface as the layers are deposited.

The two-dimensional surface contours at the Sampling lines are presented in Figure 3.12. Figure 3.12(a) shows that the Enhanced-Pixel strategy generated a slightly smoother contour, above the 11-mm quote (axis of the ordering direction of the nodes). The Zigzag strategy provided a silhouette with depression below the 10-mm. Similar outcomes were observed in the outlines of Figure 3.12(c).

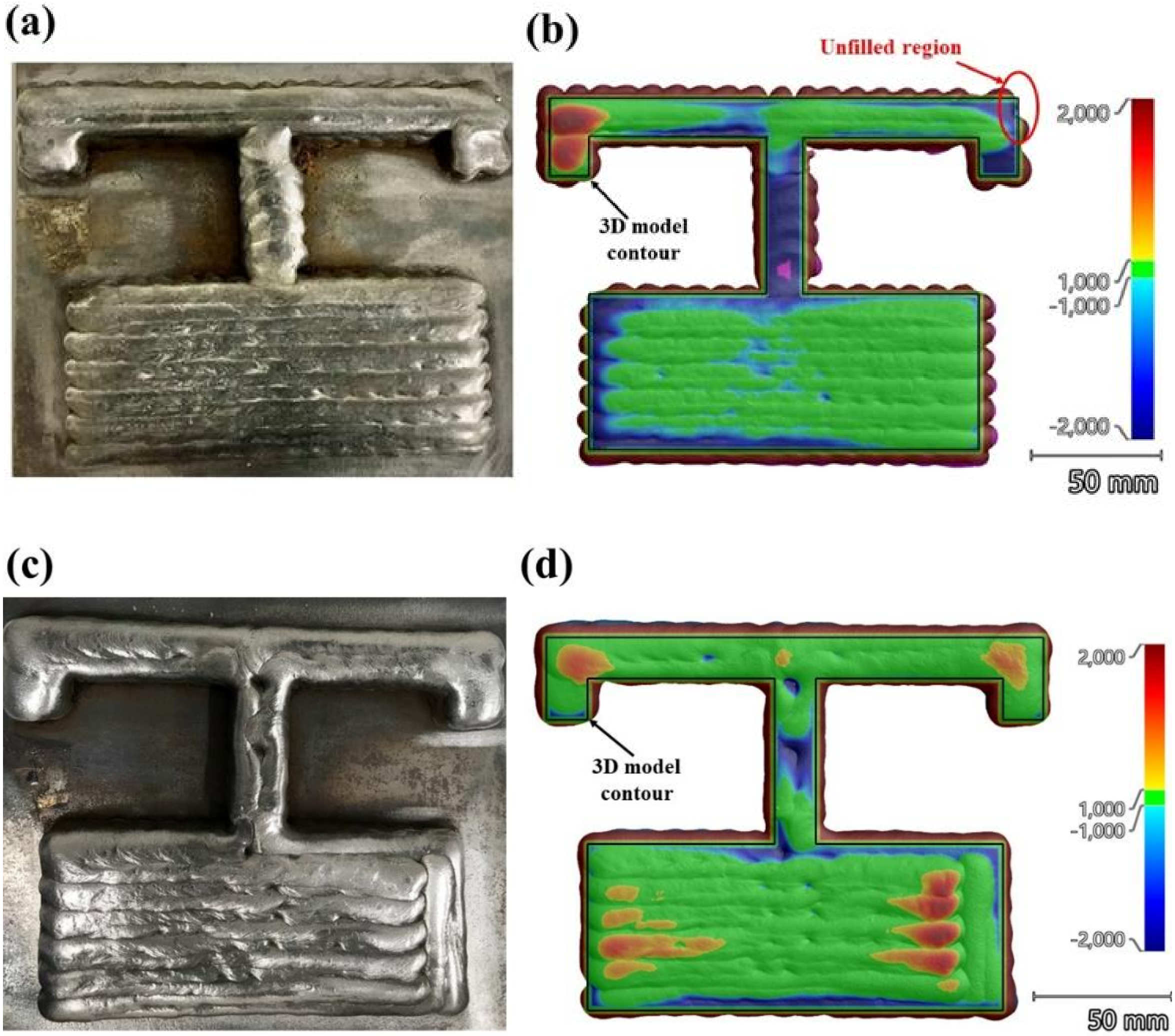
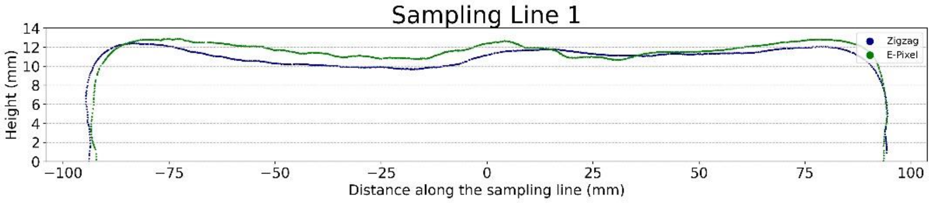
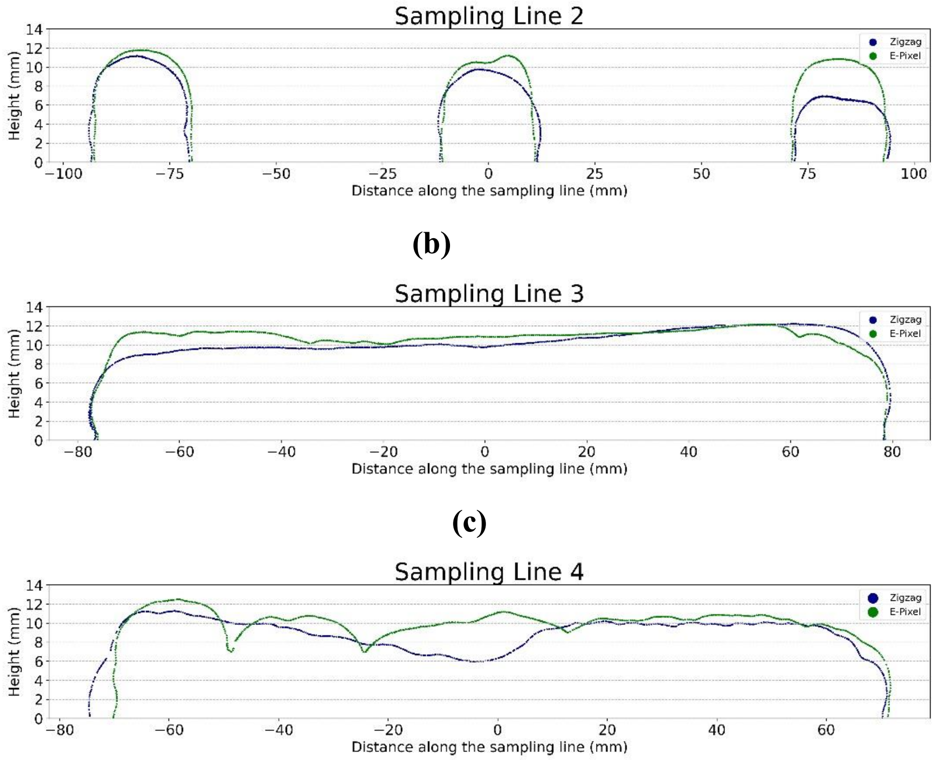


Figure 3.11 - "Part 1" and its respective scanned meshes, according to the deposition strategies: (a) and (b) with Zigzag; (c) and (d) with Enhanced-Pixel



(a)



(b)

(c)

(d)

Figure 3.12 - Two-dimensional contours at the Sampling lines defined in Figure 37 from the parts built with the Zigzag (Zigzag) and Enhanced-Pixel (E-Pixel) strategies

Based on Sampling line 2, Figure 3.12(b), there is an apparent lack of material (approximately 4 mm lower) in the rightmost region with the Zigzag strategy, precisely where the arc stopping occurred and where there would be an unfilled region due to the absence of a scan vector, as already commented. The Enhanced-Pixel strategy, in turn, does not generate these nonconformities, because it is a Space-Filling strategy, that is, they fill entirely an area and generate continuous trajectories with only one arc striking and stopping per layer. In addition, the Enhanced-Pixel strategy has the advantage of a continuous, closed-loop deposition, where the layer starts and ends overlap. With the use of the Enhanced-Pixel strategy, non-conformities occurred near the T-section of the part, which is a region prone to this failure. As seen in Figure 3.12(d), the Enhanced-Pixel strategy showed a more satisfactory contour, being its dimensional variations not below the quote of 7 mm (even with the lack of material found).

Summarising the results presented and discussed holistically, Table 3.4 shows the average top surface waviness (TSW) for the 3 sampling lines. These results indicate that the Enhanced-

Pixel presents shorter geometric deviations than the Zigzag strategy for 3 of the Sampling lines, which asserts its advantage in this case study.

Table 3.3 - *TSW* and lowest surface level from the Sampling lines

Sampling line	***TSW* (mm)**		**Lower surface value (mm)**	
	Zigzag	**Enhanced-Pixel**	**Zigzag**	**Enhanced-Pixel**
1	0.69	0.64	9.95	11.04
3	0.93	0.58	7.86	8.12
4	1.43	0.98	6.03	6.95

One must remember that not excessive surface non-conformities can be resolved with surface machining. Figure 3.13 presents the part made with the Enhanced-Pixel strategy after machining. The part built with the Zigzag strategy would be more challenging to machine (more buy-to-apply index).

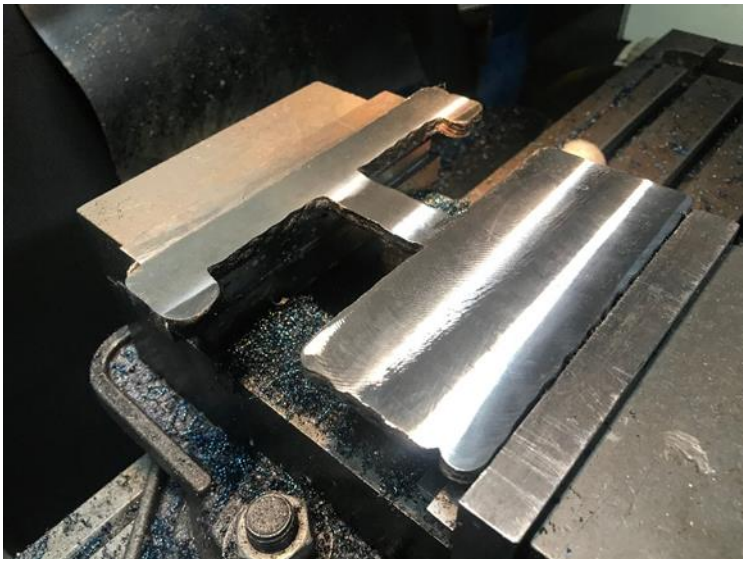


Figure 3.13 - "Part 1" built with Enhanced-Pixel strategy after top surface

3.5 Partial conclusions

The objective of this chapter was to investigate improvements to the Basic-Pixel strategy and compare its operational efficiency and effectiveness with conventional strategies. Based on the results, the following conclusions were drawn:

1. The proposed improvements to the Pixel strategy (referred to as Enhanced-Pixel), which include a new method for sorting nodes and the addition of a new heuristic for trajectory planning, positively impacted the strategy's performance in terms of trajectory distance (accomplishing the SO 3.1);
2. The random selection of the node at the beginning of the trajectory was found to be more efficient than selecting a node belonging to the periphery of the discretised nodes when using the Enhanced-Pixel strategy (accomplishing the SO 3.1);
3. The Enhanced-Pixel strategy exhibited satisfactory operational efficiency (acceptable distances and building time) and effectiveness (a lower dimensional deviation compared to the 3D model) compared to Zigzag and Parallel Contour strategies, at least in the case studies performed (with specific shapes and dimensions) (accomplishing the SO 3.2).

Therefore, the Enhanced-Pixel strategy, as proposed in this chapter, can be a candidate for building efficient and effective complex parts. In addition, considering the results obtained in the built part presented in Figure 3.13 (a shape with bulky and narrow regions), there is evidence that the Enhanced-Pixel strategy can be applied to builds with narrow regions. Therefore, an investigation will also be carried out in this sense, with more parts and with greater complexity (Chapter IV). However, it is worth mentioning that due to the flexibility of the Enhanced-Pixel strategy, many other adaptations to the algorithm can be made in order to improve operational efficiency and operational effectiveness.

As a limiting factor, even with the operator's ability to limit the search for the best trajectory (done primitively by a greedy algorithm), the improvement ended up with an increased computational time to find a good trajectory. So, further progress on the current Enhanced-Pixel strategy will focus on a smart approach for selecting the best trajectories (focus of the Chapter IV and V).

CHAPTER IV

AN ADVANCED PIXEL PATH PLANNING STRATEGY FOR WIRE ARC ADDITIVE MANUFACTURING APPLYING A REINFORCEMENT APPROACH

4.1 Introduction, Scientific Questions, and Specific Objectives

In the last chapter, a subsequent improvement of the Pixel strategy was introduced, called Enhanced-Pixel. This strategy was successfully applied to print a part with mixed characteristics (slender and bulky). The Enhanced-Pixel strategy offers advances, including the incorporation of a new axis ordering and a new trajectory planning heuristic. By utilising a double axis ordering directions (x and y) and employing five heuristics of trajectory planning, the Enhanced-Pixel strategy extensively explores the optimisation space, leading to improved outcomes. However, it is important to note that computational time can be a potential drawback for this new version of the Pixel family of strategies. This is because the algorithm needs to evaluate each of the 10 combinations of axis ordering and heuristics of trajectory planning to determine the best result for each iteration. The use of a greedy algorithm called GRASP, which is at the core of the Pixel strategy, can contribute to the time-consuming nature of this process. The algorithm's greedy nature involves assessing multiple possibilities and choosing the most promising one at each iteration. However, this approach demands significant computational resources and can lead to longer execution times. Additionally, there is a risk of generating suboptimal trajectory distances, necessitating multiple iterations to attain satisfactory results.

To overcome these new challenges, an artificially intelligent selection of the axis ordering and heuristics for trajectory planning at each iteration becomes crucial during the algorithm execution. The main goal is to determine, through artificial intelligence (AI), the overall results (the best combination that maximises outcomes), focusing on achieving the smallest trajectory distances. Considering these factors, it would be beneficial to add a machine-learning algorithm in the Enhanced-Pixel code that could adapt itself during its execution (a self-learning approach). According to Kumar et al. (2022), machine learning is the science of teaching machines to learn on their own to solve real-time problems based on input data. Machine Learning could be typified as three different techniques, namely, Supervised, Unsupervised and Reinforcement. "Supervised" machine learning is the most commonly implemented machine learning technique. Using this approach, the models need to learn functions so that inputs fit the outputs. Then, the function reveals information from categorised training data, and each intake is related to its assigned value. A large number of data is demanded to get precise results. Neural Network for face recognition is a typical example of supervised learning. Techniques classified as "Unsupervised" machine learning are capable tools for detecting similarities, thus concluding out of unclassified data by clustering them based on their similarities (restructuring the input record into new features or a set of objects with the same patterns). Searching platforms (engines) are examples of unsupervised learning usage, using clustering algorithms to group information into small numbers that are the same or associated with each other, including word frequency, sentence length, page count, etc. When there are enough data to solve the problem, Unsupervised machine learning is not likely the best option. "Reinforcement Learning" takes a different approach between supervised and unsupervised techniques. It does not rely on pre-existing data-driven relationships but instead employs a training method that rewards desired behaviours and penalises undesired ones to guide the learning process. In brief, the reinforcement learning method works on interacting with the environment, whereas the supervised learning method works on given sample data or examples.

In the context of the challenge facing, Reinforcement Learning seems to fit the enhancement of the Pixel strategy. Unlike methods that require a pre-built database of parts and shapes, the Pixel strategy could learn and adapt this means during its execution (a self-learning approach). Statements by Singh et al. (2022) and Hutsebaut-Buysse et al. (2022) support the inclusion of Reinforcement Learning in the trajectory planning process, enabling algorithms to make real-time decisions based on feedback received during execution. In conclusion, the integration of reinforcement learning empowers the algorithm to autonomously optimise its choices (a self-learning approach). According to Bouneffouf et al. (2020) and Silva et al. (2022), one prominent reinforcement learning technique, a solution for the known Multi-Armed Bandit (MAB) problem, is widely used for sequential decision-making problems where the agent selects from a set of actions, each with an unknown reward distribution. Still, Silva et al. (2022) mentioned that the MAB algorithm had demonstrated remarkable effectiveness in various real-world applications, including recommender systems, information retrieval, healthcare, and finance. Leveraging the MAB-based algorithm, the challenge of selecting (better saying recommending, to use Improvement Learning terms) the optimal axis ordering and trajectory planning heuristic during the run of the algorithm can be intelligently addressed. By employing MAB, the algorithm can dynamically adapt its decision-making process based on the feedback received from the environment (iterations). This enables the algorithm to effectively balance the trade-off between quality response and computational time, leading to improved performance and efficiency in trajectory planning (higher overall quality of the printed components while optimising the computational resources).

Given the above, a first scientific question (SQ) arises with its respective hypothesis (H):

- SQ 4.1 - How does the inclusion of the Multi-armed bandit (MAB) reinforcement learning algorithm impact the performance (trajectory distances and computational time) of the Enhanced-Pixel strategy?
- H 4.1 - Integrating MAB into the Enhanced-Pixel strategy can lead to a significant improvement in algorithm performance compared to the original strategy. By employing MAB, it is possible to identify the optimal combination of axis ordering and trajectory planning heuristic that maximizes the results relative to the Pixel strategy, resulting in shorter trajectory distances, improved convergence rate and decreased computational time.
- SQ 4.2 - How does the inclusion of the Multi-armed bandit (MAB) reinforcement learning algorithm impact the operational efficiency and effectiveness of the Enhanced-Pixel strategy?
- H 4.2 - By incorporating the MAB reinforcement learning algorithm into the Enhanced-Pixel strategy can result in a significant improvement in operational efficiency and effectiveness when compared to the original strategy. This is due to the expected performance improvement of the algorithm through the intelligent selection of the axis ordering and trajectory planning heuristic, resulting in shorter trajectory distances, for instance.

Considering the scientific question and hypothesis mentioned, the following specific objective (SO) were proposed:

- SO 4.1 - The objective is to investigate whether using a reinforcement learning algorithm, specifically MAB, in the Pixel strategy would result in improved algorithm performance.
- SO 4.2 - The objective is to investigate whether using a reinforcement learning algorithm, specifically MAB, in the Pixel strategy would result in significant improvement in operational efficiency and effectiveness when compared to the original strategy.

4.2 Literature Review: background about the optimisation tools

4.2.1 Reinforcement Learning

Learning through interaction with the environment is fundamental to our understanding of learning. Children, animals, and new employees, for instance, learn through play and exploration without the need for explicit teachers. Their daily connection to the environment provides valuable information about cause and effect, the consequences of actions, and the strategies for achieving goals. Throughout their lives, interactions with the environment serve as a significant source of knowledge about themselves and the world around them. Therefore, Reinforcement Learning is typically a learning method through interaction with the environment.

To better understand the idea behind algorithms that employ the Reinforcement Learning technique, the reader needs first to familiarise themselves with its main elements and concepts (terms), as it is proposed to do using Table 4.1.

Table 4.1 - Reinforcement Learning-related terms and their definitions

Reinforcement Learning terms	Definition
Agent	Who or what takes actions that affect the environment.
Action	The set of all possible operations/moves the agent can make.
Environment	The place where the agent gathers information, interacts with its surroundings, and acquires knowledge through learning processes.
Reward	A feedback signal provided by the environment to the learning agent (it can be positive rewards, or simply rewards, or negative rewards, or simply regrets).
State	A particular situation in which the environment is at a given moment.
Episode	A set of interactions between the agent and the environment that starts from an initial state and ends in a terminal state.

Policy	Set of rules that an agent at a given state must follow to select an action to maximise the reward and avoid regrets.
Value function	The metric used to estimate the expected return or cumulative reward an agent can obtain in a given state.
Exploration	To gather information to understand the environment better.
Exploitation	To use the information from Exploration to reach the target results.
Model-based	Type of Reinforcement Learning that constructs a representation of the environment to interact with and utilises it to simulate different actions and plan its decisions.
Model-free	Type of Reinforcement Learning that does not rely on a pre-defined model, but learns directly from interactions with the environment, only observing rewards and states

At its core, Reinforcement Learning revolves around the interaction between an agent and its environment. In other words, Reinforcement Learning problems involve an agent that learns by interacting with an environment to achieve a specific goal. Unlike other machine learning techniques, reinforcement learning provides rewards (positive or negative) for each action taken (positive rewards for good actions and negative rewards, regrets, for bad actions). In each agent interaction with the environment (called episode), the agent performs actions that return rewards or regrets. As the agent interacts with the environment, it learns to make better decisions according to a value function and policy. Furthermore, Ochi and Kamiura (2015) state that a good policy should reach a good trade-off between exploration and exploitation to maximise the reward.

To better illustrate the concepts discussed above, let one consider two practical examples, the first to demonstrate model-free reinforcement learning in training dogs and the second to showcase model-based reinforcement learning in a maze navigation scenario. In the context of training dogs, a reward-based system (model-free) is used to teach them specific commands. Through iterative trial and error, the dogs (the agent) learn to associate the commands with specific actions (for example, sitting). In a household environment (the environment), the training process involves recognising the dogs (positive rewards), typically with food, when they correctly respond to the desired action, and applying penalties or punishments for incorrect behaviours (negative rewards). This repetitive exposure to commands (episode) and reinforcement of rewards and regrets gradually shape the dogs' behaviour, leading to the desired response. It is important to note that not all concepts and elements discussed earlier are necessarily utilised in this example, as their applicability depends on the complexity of the momentary problem.

The second scenario is composed of a robot (the agent) learning to navigate through a maze (the environment). The maze comprises a grid with walls, open paths, and a goal location. In this case, a model-based approach is employed in which a model of the environment is built to simulate and plan ahead. The model captures the dynamics of the maze, enabling the agent to predict the next state and reward based on its current state and chosen action. With the model in place, the agent can employ planning algorithms to simulate and evaluate different sequences of actions. The model helps the agent determine the optimal actions to take at different states. In model-based reinforcement learning, the policy is often updated based on the planning results and the estimated values of different actions in different states (value function).

In both cases (model-free and model-based), the agent must balance exploration and exploitation to achieve success. It explores the environment by taking different actions to gather information and learn more about (the dog's personality and the maze). At the same time, it exploits its learned policy to take actions that maximise its expected cumulative reward. By balancing exploration and exploitation, the agent can effectively train the dog, navigate the maze, and reach the desired goal.

Considering that modelling each of the possible environments in a Space-Filling path planning strategy (as even the Enhanced-Pixel) would not be doable (depending on the shape of the different printable parts), it becomes evident that using a model-free method is more suitable. Among the various Reinforcement Learning methods, the Multi-Armed Bandit (MAB) algorithm is a notable model-free technique. As mentioned in the introductory section of this chapter, the MAB algorithm has gained significant attention in various real-world applications, including recommender systems, information retrieval, healthcare, and finance. The reason for this widespread adoption lies in the algorithm's suitability for tackling sequential decision-making problems that involve recommending actions from a set, where the reward distribution of each action is unknown.

4.2.2 The Multi-Armed Bandit (MAB) Problem

The denomination Multi-Armed Bandit is due to the analogy with a set of slot machines (slang for slot machine is "one-armed bandit"). Therefore, the name is not related to a robbery or a thief, yet to the probabilistic chances of winning more times using the same set of slot machines over and over. In the literature and various sources from search engines, the MAB problem is often symbolised by an octopus with each tentacle playing in one machine simultaneously (see Figure 4.1). But, this is symbolic (according to the understanding of the researchers involved in this project), because the concept of Reinforcement Learning and its derivative techniques, such as MAB problem, is based on probabilistic; accepting that different machines would have a variable award rate would be unthinkable (unless biased by cheating). The symbolism is to show the use of several alternatives at the same time, and choose the ones to continue that were given more rewards, by change (a question of luck day of a player). If it were not based on probabilistic, there would be no need to continue with the same alternatives for another day the player would play (another episode).

Figure 4.1 - A symbolic representation of Multi-Armed Bandit problem
Available in: **Towards Data Science** Acess in: 25/05/2023

A good example to justify how the MAB problem works would be the case of online advertising. Imagine a company that wants to determine the most effective product advertisement (hereafter referred to only as advertising) to display on its website to maximise user engagement and click-through rates (famous market metrics nowadays). Just to explain, clicks are a marketing metric that counts the number of times users have clicked on a digital advertisement to reach an online property. In this scenario, the company treats each product advertisement as a "slot machine" and actions in the reinforcement learning nomenclature. At the start, the company's recommender system (the agent) has limited prior information about the effectiveness of each product advertisement. The goal is to allocate the available advertising resources to maximise the user clicks (the rewards). However, the company has a fixed and limited budget or resource to allocate among the different advertisements. To solve this problem using the MAB framework, the recommender system iteratively selects advertisements based on a strategy (policy) that balances exploration and exploitation. Initially, the recommender system may allocate resources to each advertisement uniformly or randomly (exploration). At the beginning, the same probability for each advertisement to be displayed, allowing it to gather feedback and collect data on the performance of each advertisement. As the recommender system receives rewards, such as ad clicks from users, along the interactions with the environment (episode), it updates its estimate of the expected compensation or value associated with each advertisement (usually known as value function).

Based on the estimated values, the recommender system can recommend the advertisement with the highest value function (exploitation) to display to users. However, exploration mechanisms must be applied to ensure that other advertisements are given a chance to prove their effectiveness. According to Almasri et al. (2021), the most popular tools used in MAB for solving exploration-exploitation dilemmas are the ε-greedy, Upper Confidence Bound (UCB), and Thompson Sampling (TS). They help strike the right balance between exploiting the best-performing advertisement and exploring other options to discover better-performing ones potentially. In dynamic scenarios where the effectiveness of advertisements can change over time due to shifts in user preferences or market trends, more exploration is required to adapt to these changes and continuously identify the best advertisement to display. By employing the Multi-Armed Bandit approach in this advertising context, the company can optimise its resource allocation and progressively learn which advertisement yields the highest rewards. This allows the company to improve the effectiveness of its advertising campaign, increase user engagement, and maximise click-through rates, ultimately leading to better business outcomes.

After explaining the Multi-Armed Bandit (MAB) algorithm, the next section delves into the application of the MAB problem in the context of the Pixel strategy, which ultimately gives rise to the Advanced-Pixel strategy.

4.3 Advanced-Pixel Strategy

This section describes how Enhanced-Pixel was adapted to the Multi-Armed Bandit problem, giving rise to the Advanced-Pixel strategy.

4.3.1 The Multi-Armed Bandit problem in the context of the Pixel strategy

The last chapter presented the Enhanced-Pixel strategy as an improved version of the Pixel strategy family, which the basic version was presented in Chapter II. The Enhanced-Pixel strategy aims to generate the shortest trajectories while ensuring that the entire layer surface is filled. The flowchart in Figure 4.2(a) provides an overview of the strategy's steps. The process begins with the input by the user of the desired number of iterations, represented by "Enter number of iteration (x)" step. The algorithm then performs a loop where, in each iteration, a random node is chosen as the starting point for a trajectory generation, represented by the "Choose a node at random for starting" step. The nodes, which have been discretised beforehand (not shown in the flowchart), are sorted based on their x and y coordinate axis ordering (AO)s, represented by "AO [x or y]" step. Next, for each ordering along the x and y axes, the trajectory is generated compulsorily using the five heuristic methods for trajectory planning. Concomitantly, the 2-opt algorithm is applied to optimise each of the trajectories, represented by the "[Heuristic of trajectory planning symbol] +2-opt" step. In total, ten trajectories are generated and stored in a matrix per iteration, and saved into "Store values in a matrix". The algorithm then extracts the shortest trajectory distance per iteration from the matrix, the "Extract the best value from matrix" step, and stores it into another matrix, represented by "Store in a best value matrix." This matrix serves as a repository for all the good values found in each iteration. At this point, the algorithm either restarts for a new iteration, repeating the process, or proceeds to the end.

As previously mentioned in the introduction section, the Enhanced-Pixel strategy incorporates a new axis ordering and trajectory planning heuristic, allowing for extensive exploration of the optimisation space. However, the computational time required for evaluating each combination is a potential drawback. Despite potentially finding the best option for each iteration, the algorithm requires multiple iterations to converge. Therefore, a more intelligent recommendation for the axis ordering (AO) and heuristic of trajectory planning (HTP) is crucial to enhance results and achieve smaller trajectory distances.

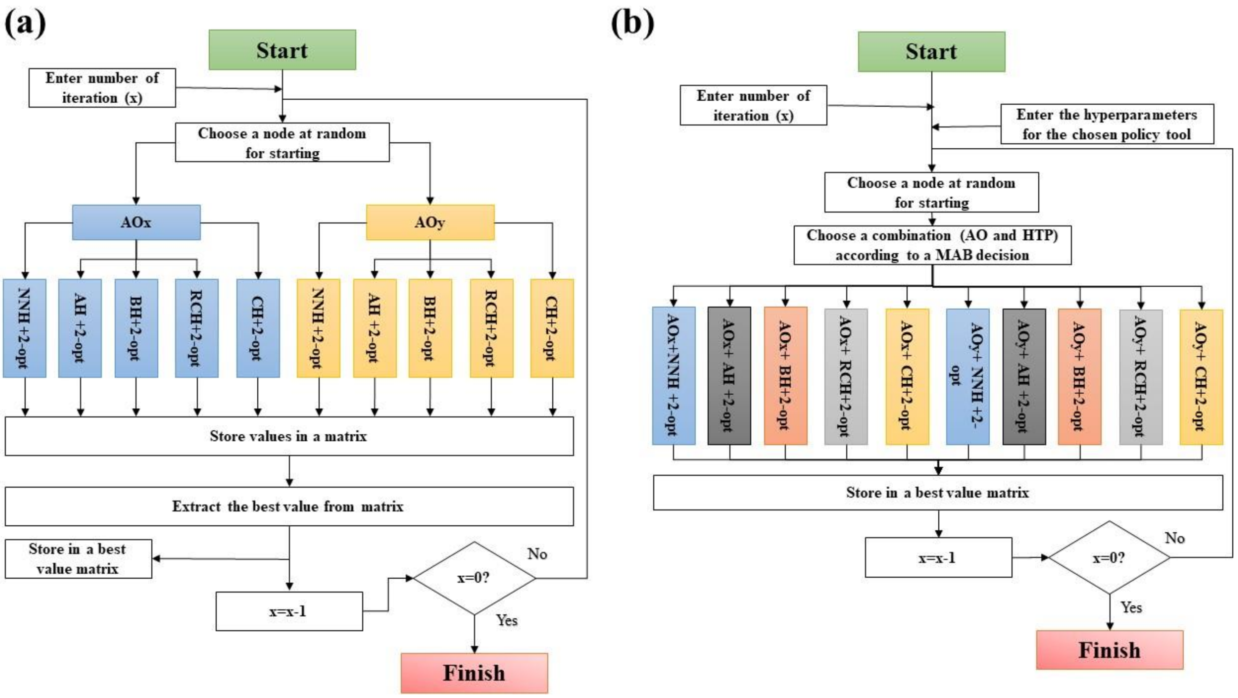

(a)
Start
Enter number of iteration (x)
Choose a node at random for starting
AOx
AOy
NNH+2-opt
AH +2-opt
BH+2-opt
RCH+2-opt
CH+2-opt
NNH+2-opt
AH +2-opt
BH+2-opt
RCH+2-opt
CH+2-opt
Store values in a matrix
Extract the best value from matrix
Store in a best value matrix
x=x-1
x=0?
No
Yes
Finish
(b)
Start
Enter number of iteration (x)
Enter the hyperparameters for the chosen policy tool
Choose a node at random for starting
Choose a combination (AO and HTP) according to a MAB decision
AOx+NNH+2-opt
AOx+ AH +2-opt
AOx+ BH+2-opt
AOx+ RCH+2-opt
AOx+ CH+2-opt
AOy+ NNH +2-opt
AOy+ AH +2-opt
AOy+ BH+2-opt
AOy+ RCH+2-opt
AOy+ CH+2-opt
Store in a best value matrix
x=x-1
x=0?
No
Yes
Finish

Figure 4.2 - Flowchart of the 2nd and 3rd generation of Pixel strategies: (a) Enhanced-Pixel and (b) Advanced-Pixel (where, NNH - Nearest Neighbour heuristic; AH – Alternate heuristic; BH – Biased heuristic; RCH - Random Contour heuristic; CH – Continuous heuristic; MAB – Multi-armed bandit framework; AO – Axis ordering, either in axis x or in axis y; and HTP – Heuristic of trajectory planning)

Following Figure 4.2(b), initially, concerning Enhanced-Pixel, the proposed algorithm described in this work follows the same steps of entering the number of iterations and selecting a random starting node. However, the user is also given the duty to choose one policy tool and its enter parameters (hyperparameters), both concepts explained in sections 4.3.1.1-4.3.1.3. The Advanced-Pixel strategy incorporates the Multi-Armed Bandit (MAB) concept by generating a single trajectory using a specific combination of the axis ordering and heuristic of trajectory planning (the action) amongst all combinations available. The combination recommendation at each iteration is guided by a chosen policy, which defines the criteria for choosing the most promising option. By utilising the MAB framework, the Advanced-Pixel strategy (the agent) eliminates the need for the conventional greedy algorithm to generate multiple trajectories (time-consuming), as the workflow of the Enhanced-Pixel version does. This strategy focuses instead on selecting the best combination, through the step "Choose a combination (AO and HTP) accordingly to a MAB decision". Once a trajectory is generated per iteration, it can be stored in the "best value matrix" using the "Store in a best value matrix" step. The steps "Store values in a matrix" and "Extract the best value from the matrix" are no longer necessary since only one trajectory is generated per iteration. At this point, the algorithm (the environment) either restarts for a new iteration, repeating the process, or reaches the end (completing the episode).

A policy is used to choose among the combinations of axis ordering (AO) and heuristic of trajectory planning (HTP) in each iteration, and it is based on the trajectory distances (rewards). While the policy selects the AO and HTP combination (exploration), but if this selection is the one with the lowest trajectory distance (value function) in each iteration, it operates from an exploitation perspective. However, a good policy should balance between exploration and exploitation. To compare to the Enhanced-Pixel strategy, the ε-greedy, Upper Confidence Bound (UCB), and Thompson Sampling (TS) tools were adapted for the Advanced-Pixel strategy. In the case of the Advanced-Pixel strategy, these policy tools have the same objective to minimise the trajectory distance. To achieve this, the value function is considered the average value of trajectory distances obtained by a given combination of AO and HTP (*h*) until that iteration *t*. The value function is represented by $Q_t(h) = (\sum_{t=1}^{n} D_t(h))/\,(N_t(h))$, where $N_t(h)$ is the number of times that *h* has been chosen until that iteration *t*, and, $\sum_{t=1}^{n} D_t(h)$ is the sum of the trajectory distance given by *h* until that iteration *t* ($1 \leq t \leq n$), where *n* is the number of iterations. The subsequent subsections provide an explanation of these policy tools in the context of the Advanced-Pixel strategy.

To clarify, it is important to note that in the subsequent equations, the negative sign in front of the value function arises from the distinction between MAB problems, where the objective is typically to maximise certain goals, and the Pixel strategy, which aims to minimise the trajectory distance. To transform a maximisation problem into a minimisation problem, a common technique is multiplying the objective function (in this case, the value function) by -1. In time, computing maximisation is usually reached using the function "*argmax*", a functional operator to find the argument that gives the class with the largest value.

4.3.1.1 ε-greedy policy tool

The ε-greedy algorithm is one of the simplest policy tools for selecting actions in reinforcement learning. The key to the success of this algorithm is the ε hyperparameter (a hyperparameter is a parameter whose value is used to control the learning process), which determines the trade-off between exploration and exploitation. Its value is defined by the user and mean the desired probability for the ε-greedy algorithm to choose each combination of AO and HTP (*h*) uniformly at random. A higher ε leads to more exploration, while a lower ε leads to more exploitation. For example, if ε is equal to 1, the choice will be completely random. On the other hand, there is a probability (1 – ε) of choosing the combination of AO and HTP (*h*) with the lowest value function (the exploitation phase) obtained up to that iteration (which we consider a good choice in this iteration). Eq. 4.1 describes the logic of the algorithm that uses ε-greedy policy.

$$h_t = \begin{cases} argmax_{h \in H}(-Q_t(h)) & \textit{with probability 1-}\varepsilon \\ \textit{choose h uniformly at random} & \textit{with probability } \varepsilon \end{cases} \tag{4.1}$$

In Eq. 4.1, h_t is a value that represents a specific combination of AO and HTP to be chosen in each iteration t, H is the set of all the ten combinations of AO and HTP. Therefore, *h* ∈ *H*. In short, the ε-greedy algorithm provides the trade-off between the exploration phase (h_t is randomly chosen) and the exploitation phase (deeper searches are performed on h_t with better results of $Q_t(h)$). However, one of the major challenges is the correct user definition of the ε hyperparameter, which manages the policy of switching between the exploration and exploitation phases of the ε-greedy algorithm.

4.3.1.2 Upper Confidence Bound (UCB) policy tool

Unlike the ε-greedy algorithm, the algorithm (tool) to develop the policy UCB use estimate uncertainties instead of random exploration to determine action selection. They prioritise actions with higher uncertainty, as they provide greater potential for information gain. This is achieved by selecting the combination of the value function, $Q_t(h)$, with the associated uncertainty $C_t(h)$, that yields the highest value. This reasoning is described in Eq. 4.2. Silva et al. (2022) explain that this uncertainty is derived from Hoeffding's inequality, represented by Eq. 4.3. This equation models the uncertainty associated with each selection of h. This uncertainty may be reduced as the value of $N_t(h)$ increases. On the other hand, the value of *t* increases every time another h is selected, which increases the uncertainty estimate. Therefore, at the beginning of the iterations, a less explored h is selected more frequently to reduce uncertainties. The use of the logarithm represents that the increase in uncertainty declines over time so that all actions will be selected. However, along with the processing time, the selection frequency will be lower for actions with a lower value estimate or that have already been elected more times.

$$h_t = argmax_{h \in H}\left(-\left(Q_t(h)\right) + C_t(h)\right) \tag{4.2}$$

$$C_t(h) = \sqrt{\frac{c * log\ t}{N_t(h)}} \tag{4.3}$$

In Eq.4.3, c is the exploration hyperparameter (it must be positive); the higher its value, the lower the confidence in the estimate. It can be noticed that when $N_t(h)$ is equal to 0, h_t is considered a maximised action (because it has not yet been explored). Therefore, at the beginning of the algorithm, the *H* must be allowed to initialise once and feed the values of $Q_t(h)$ and $N_t(h)$.

4.3.1.3 Thompson Sampling (TS) policy tool

Martin et al. (2021) emphasise that the algorithm based on the Thompson Sampling policy tool has garnered significant interest due to its effectiveness in solving the MAB problem. According to Jain et al. (2016), this policy tool was specifically developed for scenarios where the rewards from actions follow a Bernoulli distribution, i.e., they can be categorised as successes or failures. As a Bayesian method, Silva et al. (2022) state in their review that the beta distribution is a suitable model for estimating the success probability of each slot machine at each iteration, as it takes into account the number of successes and failures observed up to that point. The beta distribution is a continuous probability distribution defined on the interval [0, 1]. It is characterised by two positive shape parameters, α and β, which act as exponents of the random variable and shape the distribution. For example, as shown in Figure 4.3, when α=1 and β=1 (dashed yellow lines), a uniform distribution is obtained. This means that each slot machine (or h) will have the same probability of selection at each iteration. Dirkx and Dimitrakopoulos (2018) state that this approach is a good practice to initialise the algorithm for applying the Thompson policy tool (prior distribution), so that a good exploration phase of the heuristics occurs at the beginning.

As the iterations occur, the values of α and β are updated according to the following protocol:

1. The parameter α is incremented every time the selected h generates a trajectory with a distance lower than $Q_t(h)$ (the mean value of the trajectory distances obtained up to that iteration) in an iteration. This can be considered a good result (a success case) achieved by that selected h, so the value of α is increased. The Beta distribution with higher values of α behaves as shown in Figure 4.3 (dashed green lines), where there is a higher probability of that h being selected in the next iteration;

2. On the other hand, the parameter β is incremented for every unsuccessful iteration, that is, when the selected h generates a trajectory with a distance greater than the mean value of the trajectory distances obtained up to that iteration. This causes the distribution behaviour to be higher for low probabilities, as illustrated in Figure 4.3 (solid blue lines), which reduces the possibility of that h being selected in the next iteration.

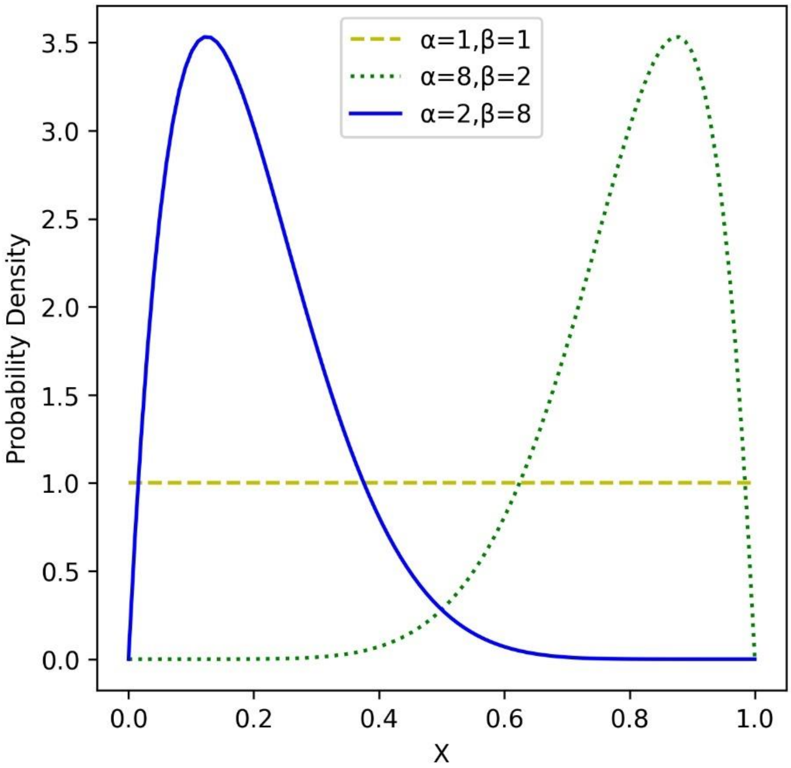


Figure 4.3 - Beta Probability density function (PDF) with varying values of the parameters α and β

Therefore, at each iteration of the algorithm, the estimation of the success probability of the h is updated. The selection of the h at each iteration *t* is based on the estimated probability that is considered optimal at that moment, as shown in Eq. 4.4:

$$h_t = argmax_{h \in H} \tilde{Q}_t(h) \tag{4.4}$$

where $\tilde{Q}_t(h)$ is the estimate of the success probability of each h, which is sampled a priori from the beta distribution.

4.4 Computational Validation

This section focuses on the first specific objective of this chapter (SO 4.1), so aims to evaluate a possible computational performance advantage of Advanced-Pixel over the Enhanced-Pixel strategy. Furthermore, various policy tools were evaluated separately within the Advanced-Pixel strategy, along with their respective hyperparameters, to determine if any of them performed worse than the Enhanced-Pixel strategy. This assessment was conducted to demonstrate the overall robustness of the Advanced-Pixel strategy, irrespective of the specific policy tool chosen. It is important to note that a single policy tool (along with its corresponding hyperparameters) must be selected to implement the Advanced-Pixel strategy in practice.

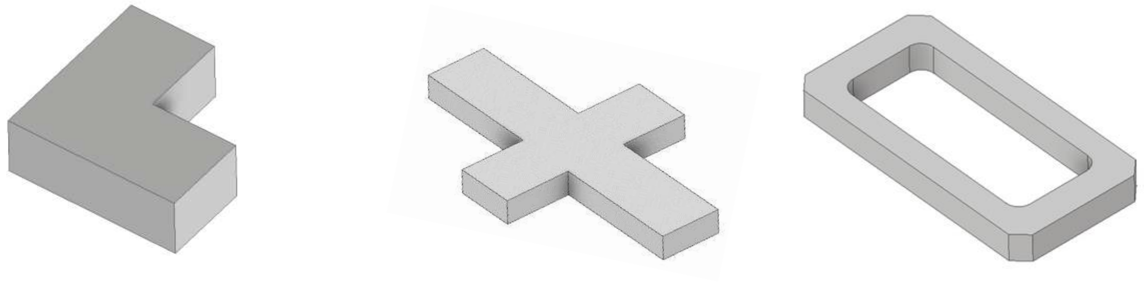
4.4.1 Methodology

Three parts with different shapes (to avoid bias), presented in 4.4 with their respective node numbers, were studied to evaluate a possible computational advantage of Advanced-Pixel (boosted with the Reinforcement Learning approach) over the Enhanced-Pixel strategy (previous version). The following policy tools (see sections 4.3.1.1, 4.3.1.2 and 4.3.1.3) and respective arbitrarily defined hyperparameters by the user were used, apart, in the Advanced-Pixel strategy (in time, a hyperparameter is used in this work as a parameter whose value is used to control the learning process):

a) ε-greedy policy tool: with ε values arbitrarily defined as 0.3, 0.5, and 1.0. However, a logical artifice was introduced in this policy tool algorithm to improve even further the tool efficiency. A decay rate was applied to the ε parametric value, so the algorithm decreases the ε hyperparameter over time at a certain rate. This artifice allows for more exploration at the beginning (with larger ε values) and more exploitation at the end (with smaller ε values). As proof of concept, a decay rate of 1% was defined. But this was applied to the case where ε was defined as 1.0 (a value that would not be reasonable if kept constant). Therefore, in this work the ε value was decreased in steps of 0.01 in each iteration;
b) Upper Confidence Bound policy tool: *c* values (hyperparameters) arbitrarily defined as 0.3, 0.5, 3.0, and 5.0;
c) Thompson Sampling policy tool: no input parameters.

Aiming at a higher performance of each tool, an extensive episode of 500 iterations was set for each path planning strategy. Considering the low number of nodes in each printable part

(Figure 4.4) and the large size of the episode, a shortage of nodes to be randomly selected by the algorithm may occur. Therefore, the trajectory initialisation node was purposely allowed to be picked more than once by the policy tool algorithm. It is important to state that the number of iterations which go through the combinations axis ordering (AO)-heuristics of trajectory planning (HTP) are not the same in Enhanced-Pixel and Advanced-Pixel. While Enhanced-Pixel evaluates 10 sets of combined AO-HTP per iteration, Advanced-Pixel evaluates only one combination per iteration. Thus, for comparison purposes, each of the 10 trajectory generations produced by a combination of AO-HTP was considered one iteration in the Enhanced-Pixel strategy. As a result, the iteration criteria turned equivalent between both strategies, by making 10 = 1 for Enhanced-Pixel and Advanced-Pixel, respectively.



245 nodes
(a)

442 nodes
(b)

630 nodes
(c)

Figure 4.4 - Different part shapes used in the computational validation and the corresponding number of nodes (remembering, the number of nodes depends on the spacing and layer area)

This study considered two criteria for the analyses: Convergence Analysis about the minimisation process of the trajectory distance and Analysis of Cumulative Regret. Convergence Analysis is standard in optimisation studies, while Cumulative Regret is commonly used for comparison between MAB algorithms (KOCH et al., 2020, and MARKÓVIC et al., 2021). In practical terms, Convergence Analysis examines the speed at which the minimal trajectory distance is achieved for each strategy over the course of iterations. It helps identify the best strategy based on minimal trajectory distances. In each iteration, a generated trajectory distance value (represented by $D_{t,h}$) is compared to the lowest value obtained thus far. If the new value is lower, it replaces the previous lowest value; otherwise, it is disregarded. Figure 4.5 demonstrates the use of Convergence curves to analyse the results of each strategy after a certain number of iterations. The orange curve (a) represents an example of a strategy, achieving a shorter trajectory (approximately 842 mm) in around 450 iterations. The blue curve (b) converges to an intermediate value of trajectory distance at around 250 iterations. In this work, the primary criterion is the trajectory distance value, while a secondary criterion is the convergence speed (lower iteration numbers). On the other hand, the green curve (c) converges faster (around 90 iterations) but does not achieve a shorter trajectory as that of the orange curve.

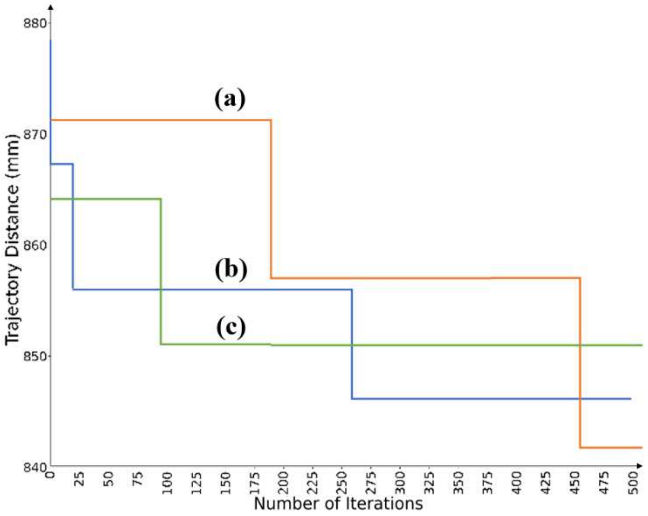


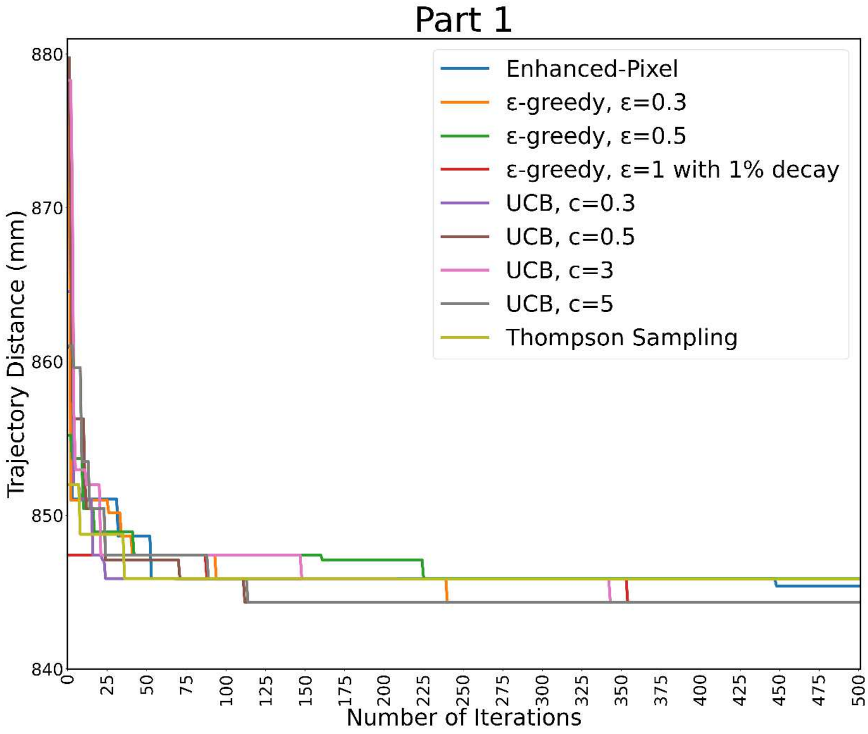
Figure 4.5 – Example of Comparative Analysis of Convergence curves for AO-HTP combinations: (a) low efficiency in convergence (long time to converge), but high efficacy (lowest trajectory distance); (b) lowest efficiency in convergence and not high efficacy; (c) highest efficiency in convergence and lowest efficacy

Cumulative Regret, the second criterion, represents the accumulated loss resulting from the chosen policy tool not selecting the optimal AO-HTP combination. Eq. 4.5 is the way to calculate the progress of this metric along the number of iterations. The ideal combination, yielding the best reward up to a given iteration (represented by D^* in Eq. 4.5), is compared to the actual combination chosen at that iteration (represented by $D_{t,h}$, Eq. 4.5). More regrets for the same number of iterations mean fewer positive rewards. Cumulative Regret tells how much one loose by betting more on the wrong combination AO-HTP; exploring more than exploiting. The analysis of Cumulative Regret curves follows a similar approach to Convergence Analysis for the minimisation process, with the distinction that regret increases as the number of iterations grows, and higher values indicate larger regrets (poor results).

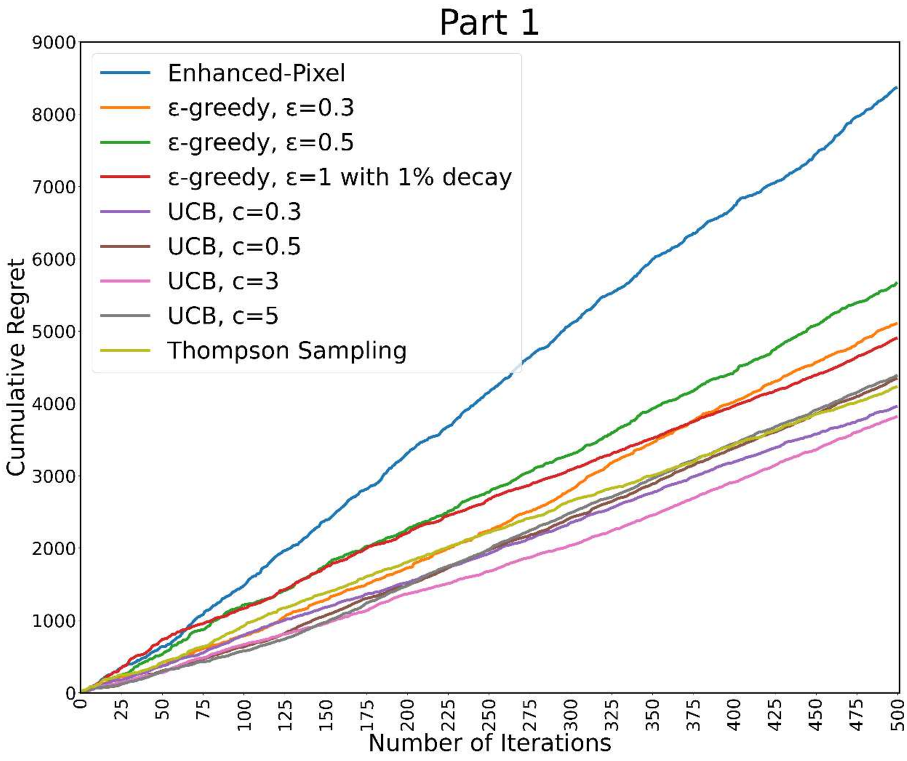
$$Regret = \sum_{t=1}^{n} [-(D^* - D_{t,h})] \tag{4.5}$$

4.4.2 Results and Discussions

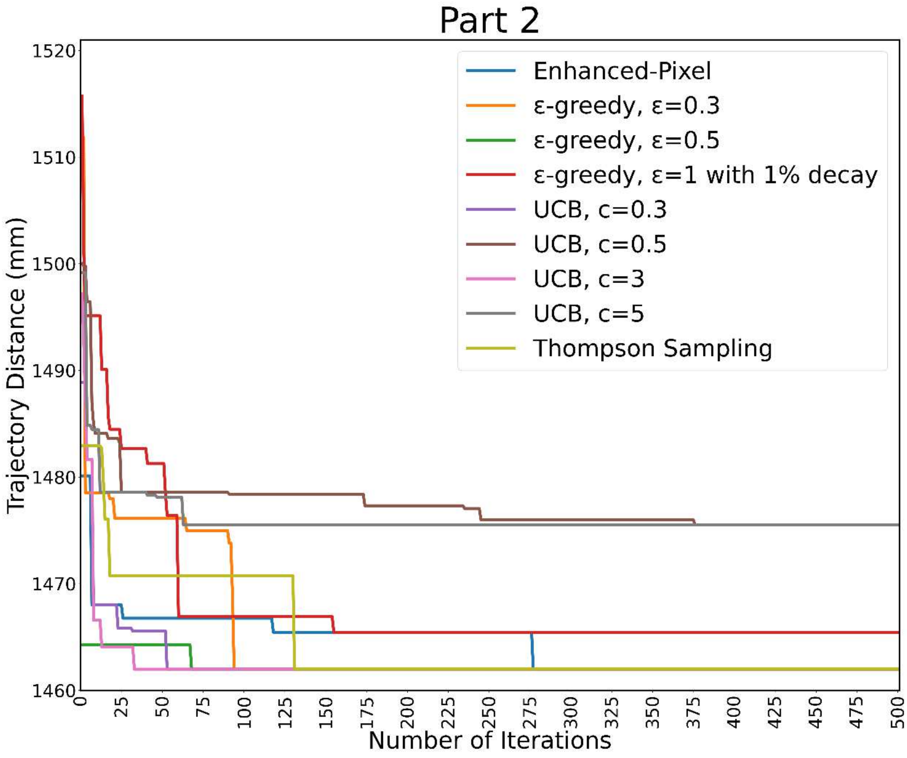
The first analysed criterion was carried out through the convergence curves. Typical corresponding curves, for the 3 case studies, are presented in Figures 4.6(a), (c), and (e). However, Table 4.2 presents the quantified trajectory distances and convergence iteration numbers for the three printed parts. The "Trajectory distance (mm) after 500 iterations" columns have highlighted in green the lowest values achieved by the specific strategies for each part. In the corresponding "Time (iterations) to converge" columns, the green highlights also represent the lowest iteration numbers among the lowest trajectory distances from each part. Conversely, the higher values are highlighted in red. Therefore, a comparison between the Advanced-Pixel and the Enhanced-Pixel strategies showed the predominance of the first strategy in all parts. However, it is important to note that not all policy tools (and their hyperparameters) used in the Advanced-Pixel strategy outperformed the Enhanced-Pixel strategy, especially in "Part 1" and "Part 2". It means that the performance of the Advanced-Pixel strategy depends on the specific policy tool and its corresponding hyperparameters. Occasionally, selecting a different policy tool may result in not-so-good performance compared to Enhanced-Pixel. One potential solution is to run the main algorithm with multiple policy tools, allowing for a comprehensive evaluation. However, it is important to note that this approach would increase the computational time required. On the other hand, in "Part 3", all policy tools applied in the Advanced-Pixel strategy consistently outperformed the Enhanced-Pixel. This suggests a trend where the Advanced-Pixel strategy, regardless of the policy tool used, may have an advantage over the Enhanced-Pixel strategy for a larger number of nodes.


Part 1
Trajectory Distance (mm)
880
870
860
850
840
Number of Iterations
0
25
50
75
100
125
150
175
200
225
250
275
300
325
350
375
400
425
450
475
500
Enhanced-Pixel
ε-greedy, ε=0.3
ε-greedy, ε=0.5
ε-greedy, ε=1 with 1% decay
UCB, c=0.3
UCB, c=0.5
UCB, c=3
UCB, c=5
Thompson Sampling

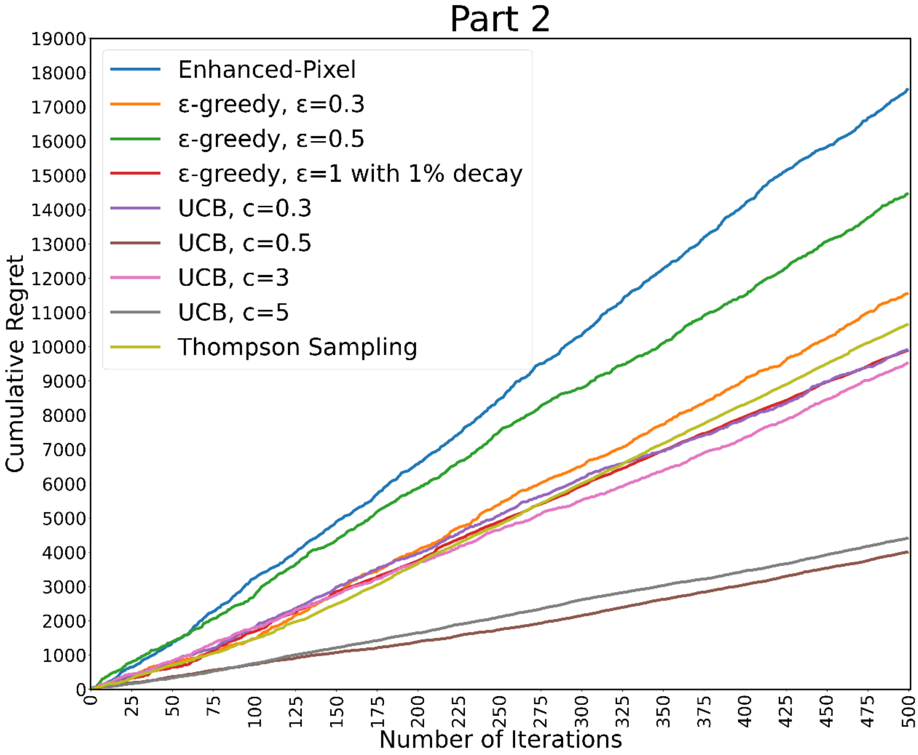
(a)


Part 1
Cumulative Regret
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
Number of Iterations
0
25
50
75
100
125
150
175
200
225
250
275
300
325
350
375
400
425
450
475
500
Enhanced-Pixel
ε-greedy, ε=0.3
ε-greedy, ε=0.5
ε-greedy, ε=1 with 1% decay
UCB, c=0.3
UCB, c=0.5
UCB, c=3
UCB, c=5
Thompson Sampling

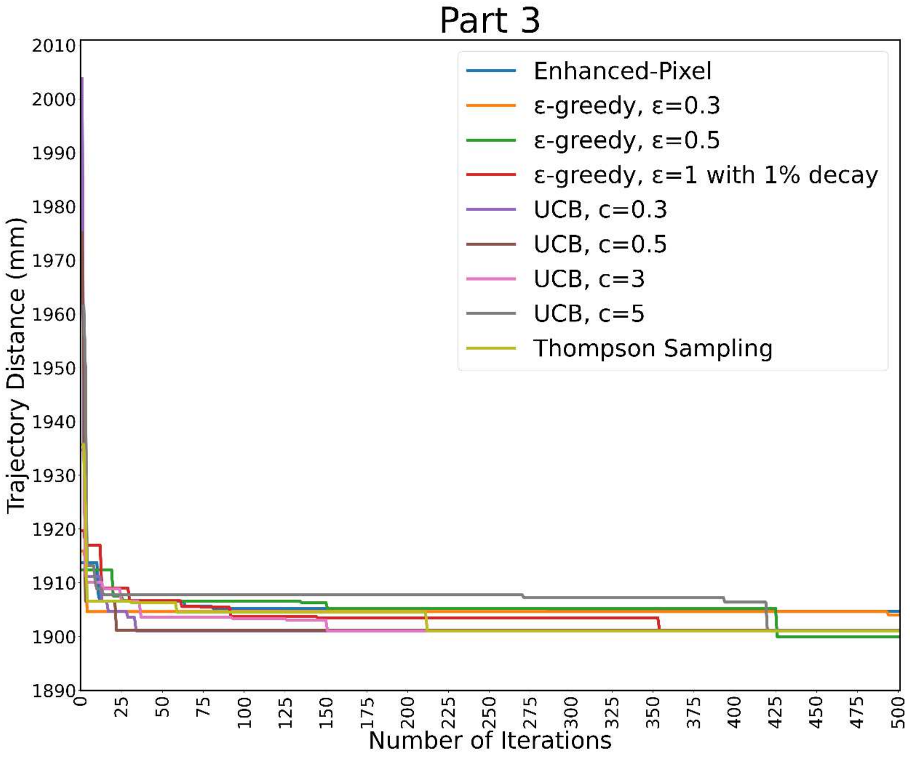
(b)


Part 2
Trajectory Distance (mm)
Number of Iterations
1460
1470
1480
1490
1500
1510
1520
0
25
50
75
100
125
150
175
200
225
250
275
300
325
350
375
400
425
450
475
500
Enhanced-Pixel
ε-greedy, ε=0.3
ε-greedy, ε=0.5
ε-greedy, ε=1 with 1% decay
UCB, c=0.3
UCB, c=0.5
UCB, c=3
UCB, c=5
Thompson Sampling

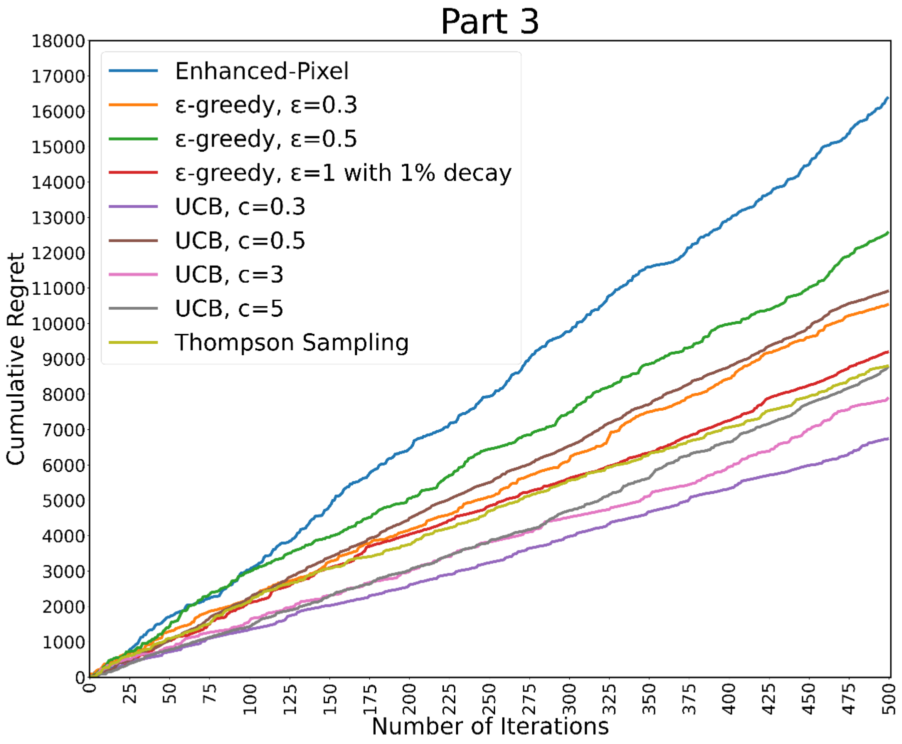
(c)


Part 2
Cumulative Regret
Number of Iterations
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
10000
11000
12000
13000
14000
15000
16000
17000
18000
19000
0
25
50
75
100
125
150
175
200
225
250
275
300
325
350
375
400
425
450
475
500
Enhanced-Pixel
ε-greedy, ε=0.3
ε-greedy, ε=0.5
ε-greedy, ε=1 with 1% decay
UCB, c=0.3
UCB, c=0.5
UCB, c=3
UCB, c=5
Thompson Sampling

(d)



(e)



(f)

Figure 4.6 - Convergence and cumulative regret charts of the trajectory plan: “Part 1” (a) and (b); “Part 2” (c) and (d); and “Part 3” (e) and (f)

Cumulative Regret was the second criterion evaluated and its curves are ilustrated in Figure 4.6(b), (d), and (e). The values of cumulative regret is presented in the third column in Table 4.2, for each part. It is evident from all three parts that the Enhanced-Pixel strategy has a higher cumulative regret (see red highlighted values). This is because it follows a non-intelligent structure that does not prioritise the best combinations of AO and HTP. Consequently, it can always yield poor results and does not learn from them. This leads to an increase in regret and makes this strategy unreliable. On the other hand, the Advanced-Pixel strategy acquires knowledge about the AO and HTP that generate the best results as the number of iterations progresses.

After evaluating the two criteria, it was found that Advanced-Pixel had better performance compared to the Enhanced-Pixel strategy. Among the policy tools (and their hyperparameters) tested in the Advanced-Pixel strategy, the UCB policy proved to be superior. In a ranking of policies tools, the UCB with hyperparameter c=3, c=0.5 and c=0.3 performed better in simulating the printing of "Part 1", "Part 2", and "Part 3", respectively (see green highlighted values for each part). It is important to note that the goal of this study was solely to verify the performance gain of the Advanced-Pixel strategy over its previous version, Enhanced-Pixel. However, it is worth mentioning that more comprehensive studies, such as using additional comparison criteria, can be conducted to determine the best policy tool within the Advanced-Pixel strategy. Based on the presented results, the "optimal" policy tool may vary depending on the specific context and the number of discretised nodes used. As mentioned earlier, it is possible to design the Multi-Armed Bandit (MAB) algorithm to accommodate multiple policies and use a policy selection mechanism to trade off between them. However, this approach should be carefully considered due to its potential impact on computational time. Future studies can explore this aspect further to find a suitable balance between comprehensive evaluation and computational efficiency.

Table 4.2 - Summary of the metrics used to assess the efficiency of the combination path planning strategy and policy tool, in a comparation between Enhanced and Advanced Pixel (where UCB stands forUpper Confidence Bound and TS for Thompson Sampling)

Strategy-tool		"Part 1"			"Part 2"			"Part 3"		
		Trajectory distance (mm) after 500 iterations	Time (iterations) to converge	Regret analysis	Trajectory distance (mm) after 500 iterations	Time (iterations) to converge	Regret analysis	Trajectory distance (mm) after 500 iterations	Time (iterations) to converge	Regret analysis
Enhanced-Pixel		845.4	450	8364.8	1461.9	277	17505.6	1904.7	75	16374.5
Advanced-Pixel	ε-greedy, ε=0.3	844.3	243	5102.61	1461.9	94	11547.3	1904.0	491	10533.7
	ε-greedy, ε=0.5	845.9	223	5661.1	1461.9	69	14459.4	1900.0	425	12565.9
	ε-greedy, ε=1 with 1% decay	844.3	355	4899.8	1465.4	154	9884.5	1901.2	353	9190.8
	UCB, c=0.3	845.9	23	3956.3	1461.9	53	9904.3	1901.1	36	6737.7
	UCB, c=0.5	844.3	111	4343.7	1475.5	378	4005.3	1901.1	22	10911.5
	UCB, c=3	844.3	341	3815.3	1461.9	31	9518.7	1901.1	150	7878.9
	UCB, c=5	844.3	113	4383.3	1475.5	63	4407.2	1901.1	421	8762.6
	TS	845.9	31	4229.4	1461.9	132	10643.0	1901.1	213	8797.4

4.5 Experimental Validation

With a specific focus on the first objective of this chapter (SO 4.2), this section aims to thoroughly assess the efficiency and effectiveness of both the Advanced-Pixel and Enhanced-Pixel strategies. The evaluation criteria for measuring efficiency and effectiveness are detailed in section 4.5.1, Methodology. The subsequent section, 4.5.2, Results and Discussions, presents a comprehensive comparison of the outcomes obtained from each strategy. By analyzing and discussing these results, we can gain valuable insights into the relative performance of the Advanced-Pixel and Enhanced-Pixel strategies.

4.5.1 Methodology

The three components shown in Figure 4.4 were printed in accordance with the experimental settings outlined in Table 4.3, using both the Enhanced-Pixel and Advanced-Pixel strategies. The number of iterations was set to 500 for both the Enhanced-Pixel and Advanced-Pixel strategies. In applying the Advanced-Pixel strategy described in this chapter as part of the experimental validation, for simplification's sake, only the best policy tool determined through computational validation (previous section) was utilized. Specifically, for "Part 1," the UCB policy tool with c = 0.5 was employed, for "Part 2," the UCB policy tool with c = 0.3 was used, and for "Part 3," the ε-greedy policy tool with ε = 0.5 was selected. Two trajectories were generated, one for the odd-layers and another for the even-layers. These trajectories were then replicated for all layers in the print, with each layer starting from a randomly selected point. The printing process involved 9 layers for "Part 1", 6 layers for "Part 2", and 4 layers for "Part 3". Printing efficiency was quantified by considering the trajectory distance and printing time, while effectiveness was assessed by examining the presence of superficial discontinuities in the printed parts, both before and after machining.

Table 4.3 - Experimental setting for printing the parts

Process	Kinects (a Cold Metal Transfer technology from Abicor Binzel)
Arc welding equipament	iRob 501 Pro (Abicor-Binzel)
Torch movement system	ABB Robot IRB 1520 ID
Substrate	SAE 1020 carbon steel (200 mm x 200 mm x 12 mm)
Substrate cooling	Natural air cooling
Wire	AWS ER70S-6 - ϕ 1.2 mm
Shielding gas	Ar + 2% CO_2 - 15 L/min
CTWD*	12 mm
Deposition speed (travel speed)	48.0 cm/min
Set wire feed speed	3.7 m/min
Set Voltage	15.2 V
Average Current	136 A

(*) CTWD – contact tip to work distance

4.5.2 Results and Discussions

Figures 4.7, 4.8, and 4.9 display the paths generated and the parts printed using both the Enhanced-Pixel and Advanced-Pixel strategies. From the top and side views, it is evident that the printed parts were shaped accordingly to the models (Figure 4.4). Given the principle of both strategies, a different starting point was randomly selected for each printed layer. These shapes of the parts may appear as too simple for the assessment. However, they are found as part of grid-stiffened panels.

Uniformly distributed surfaces indicate no material accumulation (arc starts - stops, movement accelerations and decelerations, etc.), potentially remedied by the trajectory changes between layers (odds and evens). In addition, no visible superficial discontinuities, such as voids or lack of fusion, seemed on the rough surface. However, a lower height at the edges, typical of pieces printed by WAAM, is observed from the side view. According to Li et al. (2018), this occurs due to a shortage of material caused by the near-parabolic bead shape.

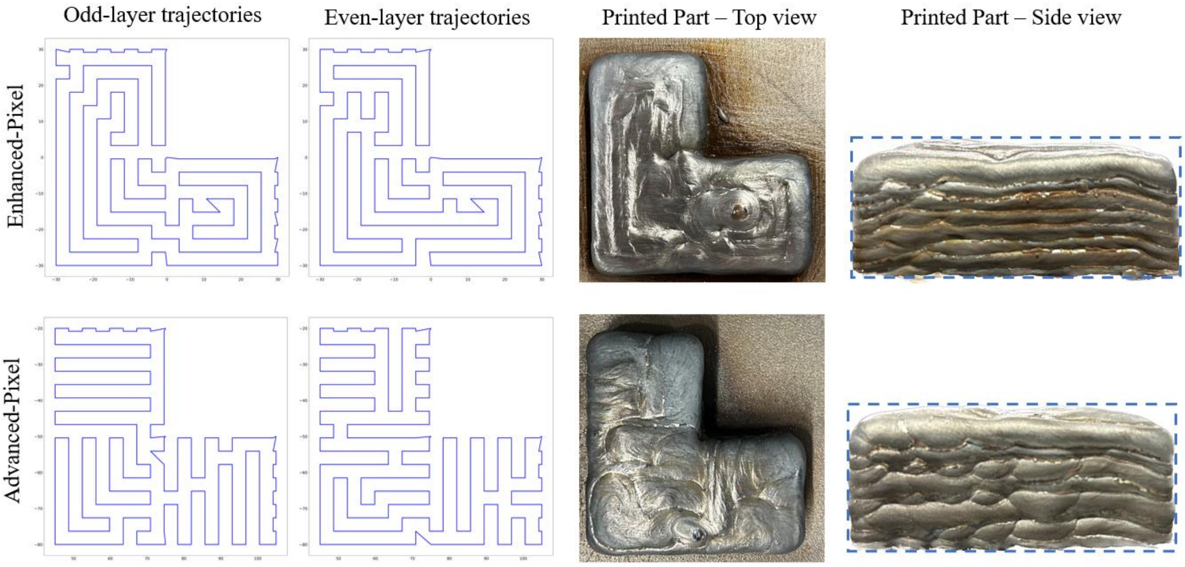


Figure 4.7 - Part 1: (Left) random trajectories designed by the planner using Enhanced-Pixel (upper) and Advanced-Pixel (lower) strategies; (centre and right) corresponding printed parts **(**trajectory starting points of the trajectory are not characterised because they were randomly selected for each printed layer)

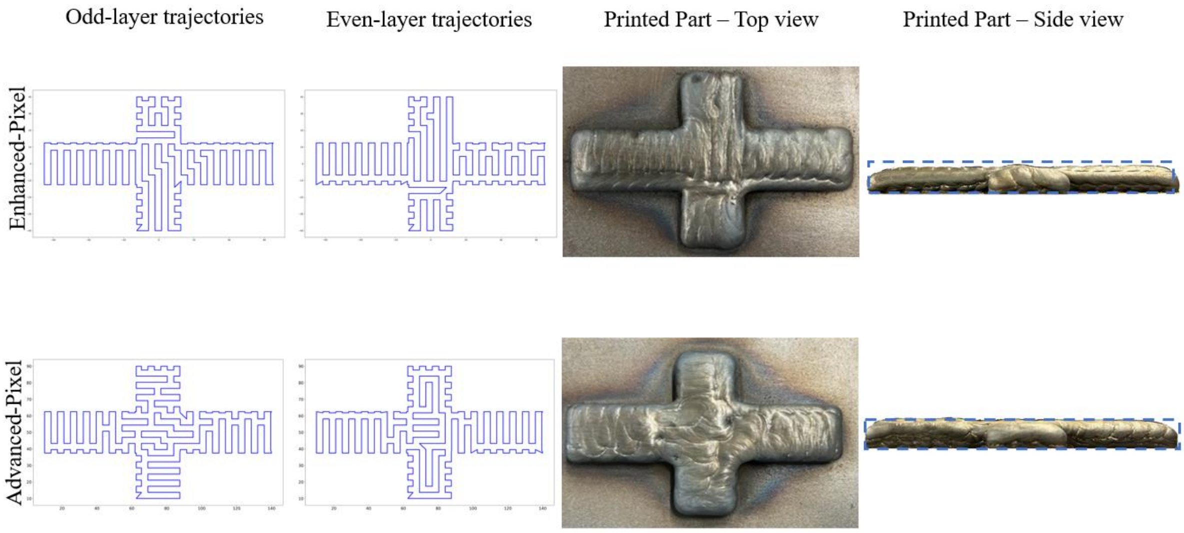


Figure 4.8 - Part 2: (Left) random trajectories designed by the planner using Enhanced-Pixel (upper) and Advanced-Pixel (lower) strategies; (centre and right) corresponding printed parts (trajectory starting points of the trajectory are not characterised because they were randomly selected for each printed layer)

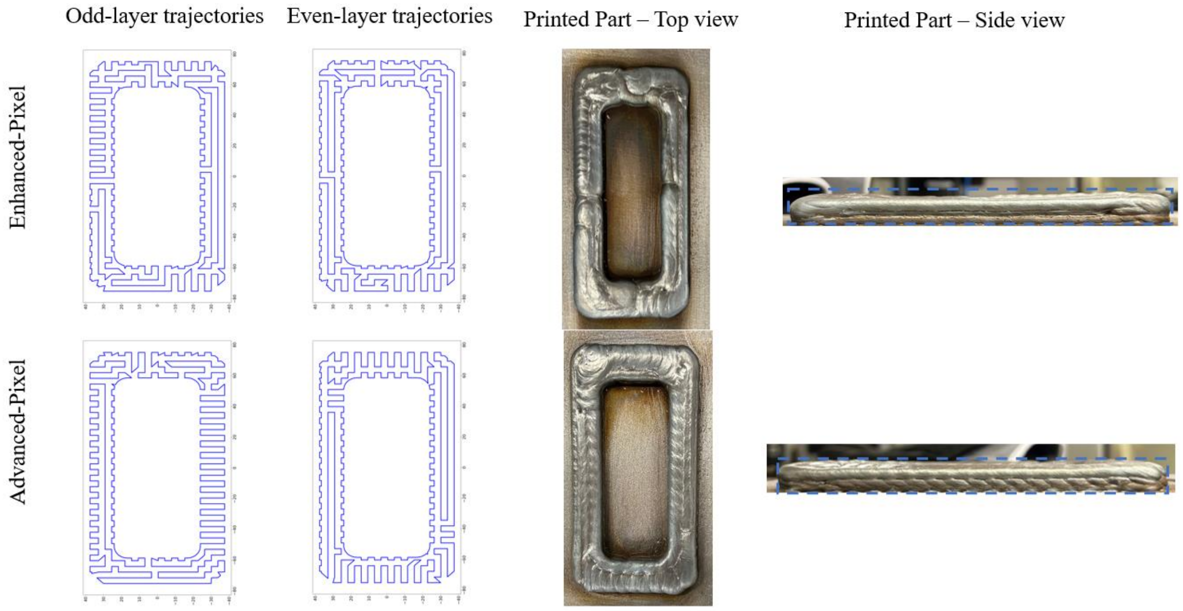


Figure 4.9 - Part 3: (Left) random trajectories designed by the planner using Enhanced-Pixel (upper) and Advanced-Pixel (lower) strategies; (centre and right) corresponding printed parts (trajectory starting points of the trajectory are not characterised because they were randomly selected for each printed layer)

Searching for discontinuities inside the parts, the upper surface of each printed part used in the experimental validation was flattened by machining (removing approximately 5 mm of material), as illustrated by Figure 4.10. The parts printed after the trajectory planned with the Advanced-Pixel strategy did not provide any visible internal discontinuity on the machined plane. However, "Part 1" and "Part 3" when printed with the Enhanced-Pixel strategy, presented some spots (resembling unfilled regions) on the same plane (marked by red circles). The cause of these erratic events was not deeply investigated, but they are hypothetically spots of improper bead overlapping. Cu et al. (2021) claim that the contact transfer controlled by current and wire feeding processes, usually characterised by low heat input, might be a contributing factor. The use of Ar-based shielding gas in this present work, with only 2% CO_2, also contributes to these events. Regardless of having at least two factors contributing to improper bead overlapping, the findings indicate that the Pixel technique, whether Enhanced or Advanced, can produce high-quality parts. But the Advanced version is prone to sounder builds.

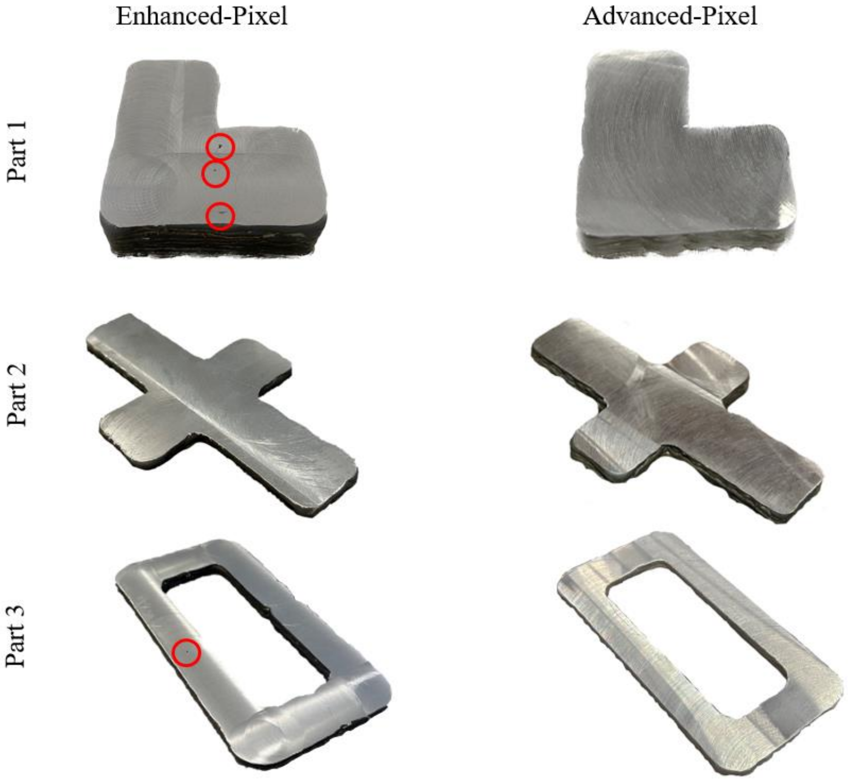


Figure 4.10 - The parts from Figures 4.7, 4.8 and 4.9 with the upper surface machined

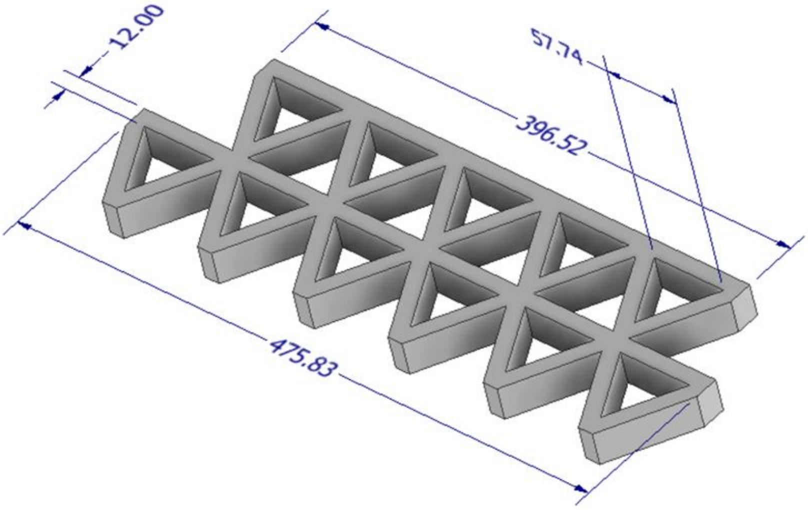
Regarding the distance of the trajectory and the printing time shown in Table 4.4, the Advanced-Pixel strategy tended remarkably to lower values (there was a tie in the printing time only in one case, namely the odd layer of Part 2). It must be noted that as the number of nodes increases (from Part 1 to Part 3), there is a tendency to increase the difference in print time between the Advanced-Pixel and the Enhanced-Pixel strategy. This may indicate that the difference in printing time would be more significant if assessed with large pieces (when the numbers of nodes are likely higher).

Table 4.4 - Printing efficiency assessment of the three printed parts (Figure 4.4)

Part Number	Layers	Criteria	Enhanced-Pixel	Advanced-Pixel
1	Odd	Trajectory Distance (mm)	845.81	845.56
		Printing time (s)	125.07	124.36
	Even	Trajectory Distance (mm)	846.45	845.56
		Printing time (s)	125.11	124.35
2	Odd	Trajectory Distance (mm)	1462.67	1462.15
		Printing time (s)	215.19	215.19
	Even	Trajectory Distance (mm)	1463.58	1462.33
		Printing time (s)	215.19	214.82
3	Odd	Trajectory Distance (mm)	1903.96	1903.39
		Printing time (s)	289.01	284.14
	Even	Trajectory Distance (mm)	1902.91	1902.33
		Printing time (s)	288.49	284.10

4.6 A Case study of a thin wall structure

The three parts (Figure 4.4) that validated the Advanced-Pixel strategy are typically bulky shapes with propositional features to qualify the performance comparison. However, there are also shapes prone to be wire arc additively manufactured that are composed of thin walls, such as angled grid structures (or lattice structures). These structures, according to Tao and Leu (2016) possess many superior properties to solid material and conventional structures. And they are feasible by additive manufacturing techniques, including WAAM. It is important to state that the trajectory planning for thin walls uses different concepts than those with bulky parts. Therefore, to further demonstrate the efficacy of the Advanced-Pixel strategy, an angled grid structure, which a model example is presented in Figure 4.11, was used as a case study.

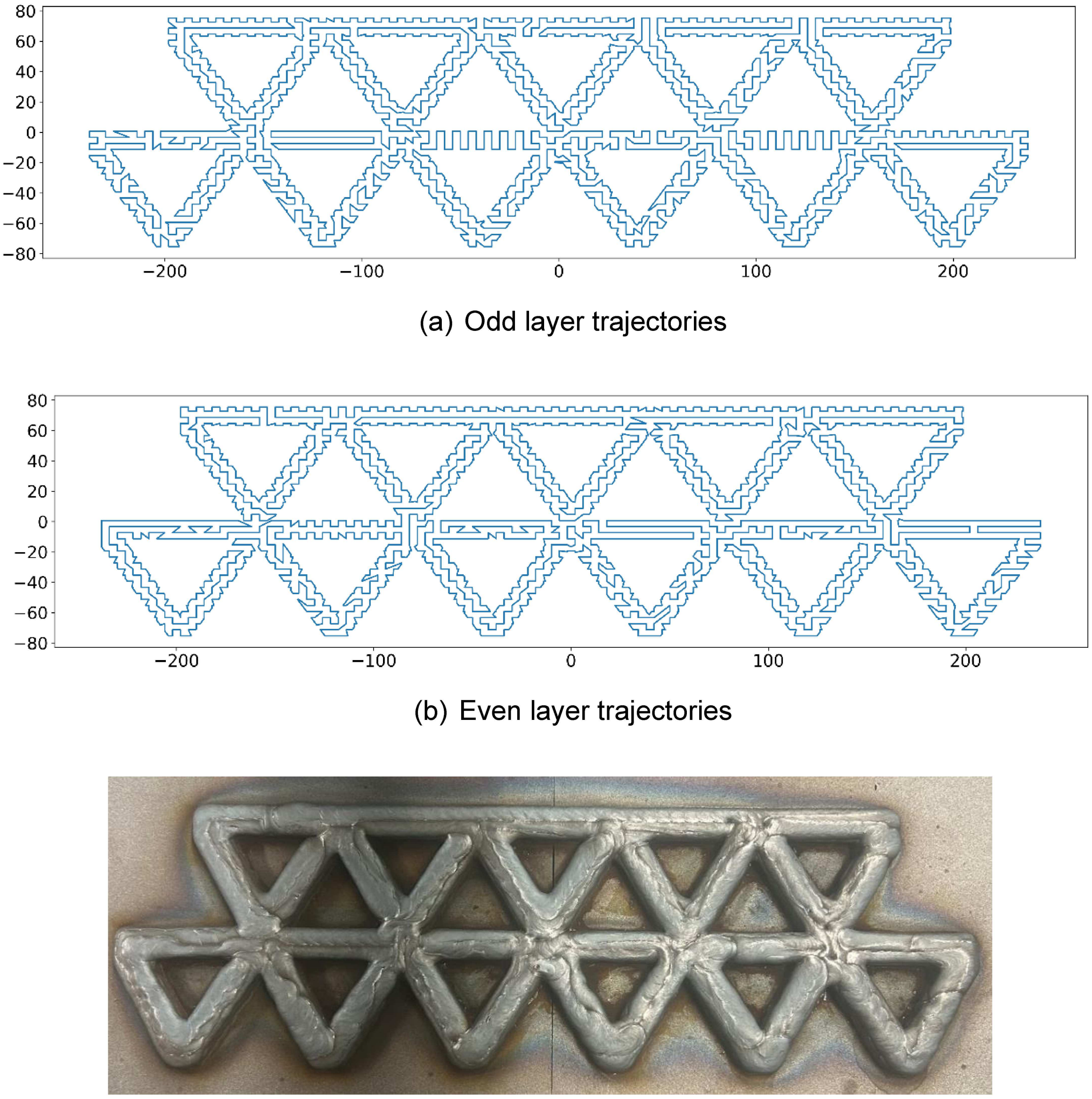


2919 nodes

Figure 4.11 - An angled grid 3D model composed of 12-mm-thick walls (dimensions in mm) with the number of nodes defined based on the spacing and layer area

The same parameters presented in Table 4.3 were used for this case study. The spacing between nodes was set to 4 mm, resulting in 2919 nodes, and the path trajectory planner performed only 20 iterations. Six layers were built up. The Thompson Sampling policy tool was deliberately chosen (just a reminder, the user selects the policy tool as an input) because of the no need for any specific hyperparameters, providing a simple but powerful method. That being said, no attempt to optimisation of the trajectory planning by using the other policy tools was made. The experimental results are shown in Figure 4.12, where continuous trajectories were generated for the odd and even layers, as seen in Figures 4.12(a) and (b). Notably, the printed part exhibits no visible voids or lack of fusion, as evidenced by Figures 4.12(c) and (d). A non-conformity was identified in the first layer, as indicated by the red circle in Figure 4.12(e). This non-conformity could be attributed to either lower energy deposition in the first layer over the substrate or to the use of two plates as substrates placed side-by-side. Additionally, cracks were observed, as indicated by the yellow circles, which are likely a result of residual stress in this region before cutting the part.

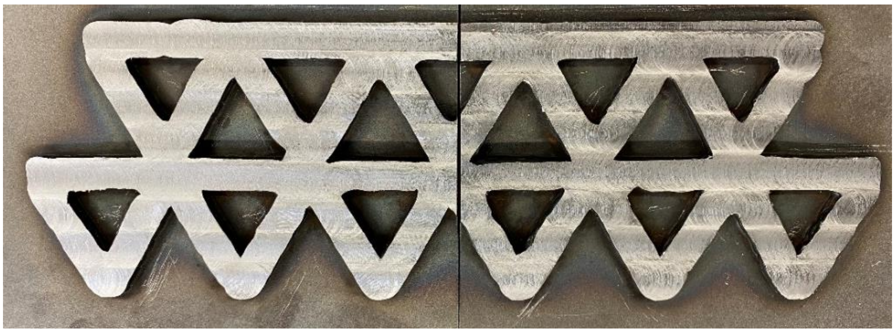
However, those discontinuities are related to the interface printed part-substrate, not the building itself.



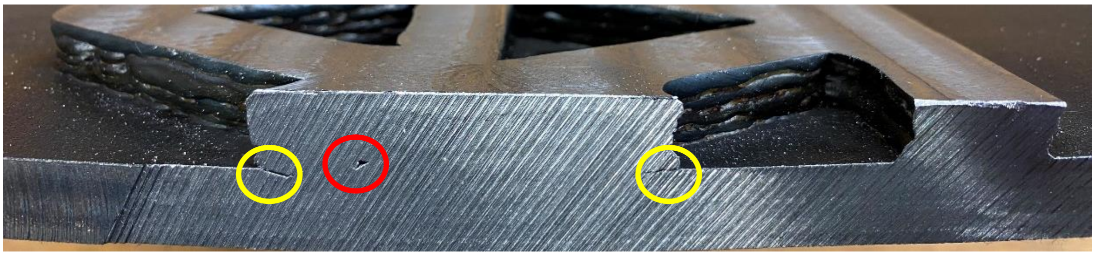
(a) Odd layer trajectories

(b) Even layer trajectories

(c) Printed part



(d) Machined surface (top view)



(e) Machined surface (side view)

Figure 4.12 - An angled grid structure: (a) and (b) - random trajectories from Advanced-Pixel strategy; (c) - corresponding printed part; (d) and (e) - top and side views, respectively, surfaces after flattened by machining

The same case study was used to confirm the higher performance of the Advanced-Pixel strategy to generate shorter trajectories and minimise printing times when compared to the Enhanced-Pixel strategy. As this could be done computationally, no printing was carried out using Enhanced-Pixel. Using a similar concept of the validation assessment (section 4.4), the Enhanced-Pixel was configured with only 2 iterations (corresponding to 20 iterations of the Advanced-Pixel strategy). Table 4.5 allows the comparison of the two strategies. The outcomes indicate that the Advanced-Pixel can generate significantly shorter trajectories and printing times than the Enhanced-Pixel. It is important to know that the angled grid model (Figure 4.11) was larger than those used in the validation stage (Figure 4.4), needing more nodes to be composed for the planner (more sensitive comparison). A significant difference of about 100 mm and about 60 s was found for trajectory distance and printing time, respectively, which was not in the results presented in section 4.5.2, as the number of nodes did not exceed 700.

Table 4.5 - Computational efficiency comparison to print an angled grid structure by enhanced and advanced Pixel strategies

Part	Layers	Criteria	Enhanced-Pixel	Advanced-Pixel
Anglegrid (Figure 4.11)	Odd	Trajectory Distance (mm)	9525.08	9420.39
		Printing time (s)	1609	1523
	Even	Trajectory Distance (mm)	9546.08	9422.64
		Printing time (s)	1620	1556

4.7 Partial conclusions

The concept of the MAB problem in the algorithm was applied to the space-filling-like Enhanced-Pixel strategy to optimise the trajectory, as stated in the objective of this work. The proposed Advanced-Pixel algorithm checks only one of the ten axis ordering (AO) plus heuristics of trajectory planning (HTP) combinations per iteration. In contrast, Enhanced-Pixel checks each of the ten combinations of AO and HTP per iteration. In summary, it was noted that:

- The algorithm performance gain shows that Advanced-Pixel converges, in most cases, to the shortest trajectory, even with shorter printing time, the latter due to fewer iterations than its predecessor (Enhanced-Pixel). However, it is worth noting that the solution applied in Advanced-Pixel is based on probabilistic concepts, and one cannot expect the advanced version to beat the predecessor in 100% of cases (accomplishing the SO 4.1);

- The sensibility of the algorithm performance comparison increases for larger printable parts (higher number of nodes)

- Concerning quality gain, the experimentally printed shapes and sizes did not show significant differences when using both strategies (Enhanced-Pixel and Advanced-Pixel) – the advanced version certainly does not impair the application of the algorithm to plan trajectories (accomplishing the SO 4.2);

- Therefore, the implementation of Reinforcement Learning through the MAB problem succeeded well towards "grading up" the Pixel family of Space-Filling trajectory planners.

Hence, the Advanced-Pixel strategy, introduced in this chapter, shows promise as a viable approach for manufacturing intricate parts such as angle grids with enhanced efficiency and effectiveness. However, it should be noted that the computational time can be prolonged depending on the number of iterations required. In order to address this limitation and further enhance both computational efficiency and response quality, the upcoming chapter will explore the concept of clusterization nodes. This innovative approach aims to optimize the computational time while maintaining high levels of response quality.

CHAPTER V

FAST ADVANCED-PIXEL STRATEGY FOR WIRE ARC ADDITIVE MANUFACTURING

5.1 Introduction, Scientific Questions, and Specific Objectives

According to the previous chapter, the successful implementation of the Multi-Armed Bandit (MAB) problem in the Enhanced-Pixel strategy has showcased its effectiveness in enhancing the performance of the Pixel strategy algorithms. By incorporating reinforcement learning through the MAB problem, the Advanced-Pixel strategy outperforms its predecessor, Enhanced-Pixel, with shorter trajectory distances and printing times. Nevertheless, it is diagnosed that the computational time increases exponentially, depending on the number of iterations required for optimization.

One way to address this issue is the utilization of clustering techniques. Clustering is a powerful machine learning technique that groups data based on shared features, offering opportunities for more efficient problem-solving. For instance, in the Travelling Salesman Problem (TSP), which exhibits exponential time complexity with increasing city count, the divide-and-conquer approach can be applied. This approach involves dividing large TSP instances into smaller clusters with fewer cities, resulting in notable improvements in both response quality and computational time. By reducing the dimensional space of optimization problems, clustering algorithms enable more effective exploration and exploitation of problem characteristics, leading to optimized solutions in a shorter timeframe.

Given the above, a couple of scientific questions (SQ) arise with their respective hypothesis (H):

- SQ 5.1 - How can a clustering technique be effectively applied to optimize the performance of the Advanced-Pixel strategy in terms of solution quality and computational time?
- H 5.1 - The use of clustering technique in the Advanced-Pixel strategy can result in improved convergence behaviour, as it enables the algorithm to focus on specific clusters

and exploit local search spaces, leading to faster convergence (reduction in computational time) to near-optimal solutions (improvement in the response quality).

- SQ 5.2 – Would apply a clustering technique in the Advanced-Pixel strategy compromise the quality of printed parts?
- H 5.2 - Implementing a clustering technique in the Advanced-Pixel strategy cannot negatively impact the quality of printed parts. In fact, clustering techniques are expected to improve the quality of the printed parts by allowing for better optimization of trajectories within specific clusters.

Considering the scientific question and hypothesis mentioned, the following specific objective (SO) were proposed:

- SO 5.1 - The objective of the work described in this chapter was to investigate the impact of applying a clustering technique to the Advanced-Pixel strategy in terms of computational time and objective function (minimising the trajectory distance). The study aims to determine whether the use of a specific clustering technique enhances the convergence speed of the Advanced-Pixel strategy, resulting in reduced computational time, while also improving the quality of the obtained solutions. Furthermore, it seeks to maintain the achieved part quality reached in the previous chapters.

5.2 Literature Review: K-means Clustering technique

Clustering is a widely used technique in machine learning that enables data grouping based on similarities in their features. Clustering algorithms can be applied to reduce the dimensional space of optimization problems, resulting in improved solution quality and computational time performance. For instance, solving the Travelling Salesman Problem (TSP), a well-known optimization problem, requires an algorithm in which the processing time grows exponentially with the number of cities. In fact, the number of possible solutions, i.e., permutations of cities, is given by *(n-1)!/2*, where *n* represents the number of cities. However, by using the called Divide and Conquer paradigm, large TSP instances can be split into clusters, with small values of *n*. This approach can significantly reduce the computational time required and also enhance the quality of solutions.

The concept of clusterization applied to solve the Travelling Salesman Problem has been largely studied as a problem solver. For instance, the Generalized Travelling Salesman Problem (GTSP), to find the minimum cost round trip that visits exactly one node from each cluster. GTSP has been applied even in manufacturing. In this line, Hajad et al. (2019) proposed an optimization algorithm for laser cutting that minimizes heat accumulation based on a Genetic Algorithm and a clustering technique to solve the GTSP. In these cases, the clusters are the contours of the part, and the idea was to visit only one node at the contour of each cluster. So, when the laser tool visits one node of a specific cluster, it processes the cutting task and moves to visit another cluster. There is still the Clustering Travelling Salesman Problem (CTSP), also called Clustered Travelling Salesman Problem. In contrast to GTSP, CTSP deals with pre-defined clusters and requires visiting all nodes within each cluster. A manufacturing-related example of CTSP is the drilling of different types of holes on a rectangular sheet using tools mounted in a carousel, as described by Laporte and Palekar (2002). The problem is modelled as a CTSP, with clusters consisting of all hole locations requiring the same tool. Researchers have studied this problem to optimize the sequencing of tools used in the same pass of drilling operations.

Therefore, the main idea to be presented in this chapter is to utilise clustering techniques to mitigate large-scale, becoming several small problems to gain intrinsic advantages. An example demonstrating the benefits of this strategy is presented in Avşar and Aliabadi (2015), who demonstrated the effectiveness of a Parallel Divide-and-Conquer approach for solving the TSP. Their approach involved dividing cities (which symbolise the nodes) into smaller regions called municipalities, and finding the most suitable solution from each municipality to obtain the best overall solution. The final solution is obtained by combining neighbouring municipalities using a blend operator. The approach outperformed standard TSP test problems, Travelling Salesman Problem Library (TSPLIB), in terms of both solution quality and time.

Among the many clustering techniques available, K-means is a powerful technique that can significantly reduce the dimensionality of large datasets, making it possible to solve previously infeasible problems. It is particularly useful for grouping similar data points together, which can help identify patterns and make predictions. In K-means clustering, *k* signifies the number of clusters (groups) wanted to form. In TSP problems, K-means can be applied to group cities that are close to each other, simplifying the process of finding a near-optimal solution. Moreover, the K-means clustering algorithm is computationally efficient and can handle large datasets with a reasonable amount of memory.

The origin of K-means clustering is somewhat uncertain. Jain (2010) describes that it was proposed for different scientific fields. According to him, it was first unclosed in 1956 by the Polish mathematician Hugo Steinhaus. But not far later, in 1957, it was used by Stuart Lloyd for PCM signal quantization. However, the technique was published only in 1982. Still, according to the authors, individually Ball and Hall rediscovered the algorithm in 1965, and MacQueen in 1967, as a clustering technique. According to Celebi et al. (2013), the K-means is a widely used unsupervised clustering technique with various applications in data mining, computer vision, and natural language processing. The literature on the K-means algorithm highlights its many practical applications, including market segmentation, fraud detection, image segmentation, document clustering, genetic clustering, and engineering (these are just a few examples of how the algorithm can be used for unsupervised clustering). In general, K-means can be applied to any dataset where the objective is to group data points into distinct clusters based on their similarity or distance.

When the concept is applied as an algorithm, K-means is an iterative clustering algorithm that aims at dividing a given dataset into *K* distinct, non-overlapping subgroups or clusters. Each data point in the dataset is assigned to only one cluster based on the nearest cluster centroid, which is the arithmetic mean of all data points belonging to that cluster. Suppose the dataset consists of *N* number of points, $X_i = \{x_1, x_2, x_3, \dots, x_N\}$ and *K* number of clusters (previously pre-defined), $C_j = \{C_1, C_2, C_3, \dots, C_K\}$. The clusters of centroids are represented by $c_j = \{c_1, c_2, c_3, \dots, c_K\}$, where c_j denotes the centroid of the cluster C_j. In general, the algorithm tries to minimize the sum of the squared distances between each data point (x_i) and its assigned cluster centroid (c_j), while also maximizing the difference or dissimilarity between the clusters. The convergence criterion (CC) is presented in Eq. (5.1).

$$CC = min \sum_{j=1}^{K} \sum_{i=1}^{N} F_{i,j} \left\| x_i - c_j \right\|^2 \tag{5.1}$$

This function is subject to Eq. (5.2).

$$F_{i,j} = \begin{cases} 1 \; if \; data \; point \; i \; belongs \; to \; cluster \; j \\ 0 \; otherwise \end{cases} \tag{5.2}$$

The K-means algorithm follows the following steps:

1. Specify the number of clusters *K*;
2. Randomly select *K* data points as the initial cluster centroids;
3. Assign each data point to the nearest cluster centroid;

4. Recalculate the cluster centroids based on the mean of all data points assigned to each cluster;
5. Repeat steps 3 and 4 until convergence, i.e., until the cluster assignments and centroids no longer change significantly.

Figure 5.1 provides a visual representation of the step-by-step process of the K-means clustering algorithm, offering a clearer understanding of its functioning.

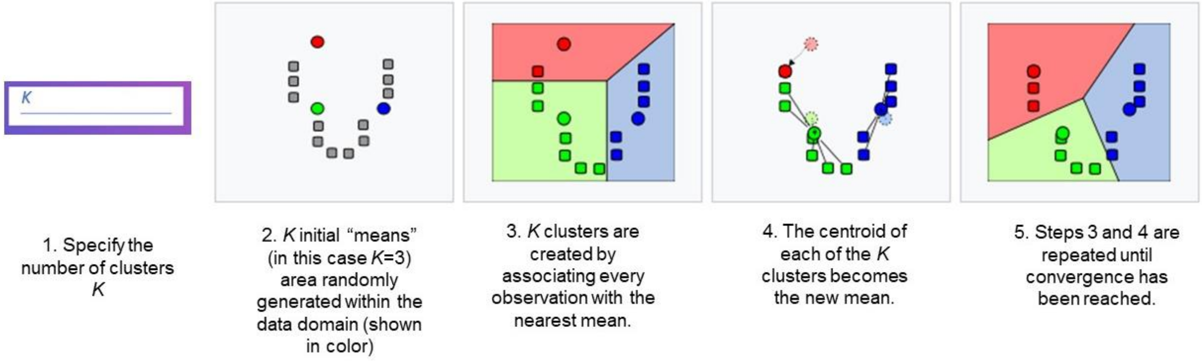

K
1. Specify the number of clusters *K*
2. *K* initial "means" (in this case *K*=3) area randomly generated within the data domain (shown in color)
3. *K* clusters are created by associating every observation with the nearest mean.
4. The centroid of each of the *K* clusters becomes the new mean.
5. Steps 3 and 4 are repeated until convergence has been reached.

Figure 5.1 - Demonstration of the standard k-means algorithm (Adapted, Wikipedia)
Fonte: https://en.wikipedia.org/wiki/K-means_clustering, Acessado em: 11 de maio de 2023

Although the K-means algorithm requires the number of clusters *K* as user input, there are several methods that can be used to optimize its selection. Two of the most popular methods are the Elbow method and the Silhouette method. Shi et al. (2021) explain that the Elbow method involves plotting the within-cluster sum of squared errors (SSE) for different values of *K*, and selecting the value of *K*, where the rate of decrease in SSE slows down and forms an elbow-like shape. On the other hand, Muca et al. (2015) explain that the Silhouette method involves computing the average silhouette width for each value of *K*, and selecting the value of *K* that maximizes the average silhouette width. The silhouette width measures how well each data point fits into its assigned cluster compared to other clusters. In addition to the Elbow and Silhouette methods, other techniques for determining the optimal number of clusters in the K-means algorithm have been proposed, such as the gap statistic technique, introduced by Tibshirani et al. (2001).

In addition to the challenge of selecting the optimal number of clusters, Xie et al. (2019) described several other limitations of the K-means algorithm, such as the assumption that clusters are spherical with equal size and density, the algorithm may not be suitable for clustering non-linear or non-convex data (i.e. spiral-shaped data), sensitive to outliers and sensitivity to initial conditions. To solve these problems, some studies have been done. For example, various studies have been conducted to address the challenges posed by sensitivity to initial conditions. For example, Fränti and Sieranoja (2019) proposed an approach that involved improving the initialization process and executing multiple runs of K-means algorith using different initial solutions. However, the authors stated that although numerous initialization techniques have been proposed in the literature, there is no clear consensus on the best approach. In light of this ambiguity, using random data points as centres in Step 2 was deemed a straightforward solution. It is always important to consider these limitations and apply the k-means algorithm appropriately based on the characteristics of the data being analyzed.

In Step 3 of the K-means clustering process, it is important to consider the characteristics of the data and the research purpose. According to Wu et al. (2021), the choice of distance measurement method directly impacts clustering results, particularly for non-linear or non-convex data. Therefore, selecting an appropriate distance metric is crucial in determining cluster properties. Numerous distance measurement methods are available, including Euclidean distance, Manhattan distance, Chebyshev distance, Cosine distance, and Correlation coefficient, among others. However, the most commonly used metrics are Euclidean distance and Manhattan distance. Regardless, it is recommended, in general, to try multiple distance metrics and compare the results to make an informed decision.

Considering all the aforementioned information, Zhao et al. (2018) described the K-means algorithm as computationally efficient, scalable, easy to understand and implement, applicable to various data types and customizable. Because K-means clustering techniques are widely adopted in machine learning and data science, this study considered their utilisation. However, it is important to observe that there are several other clustering techniques available. For instance, Subasi (2020) highlighted alternative clustering algorithms, including Hierarchical clustering, Density-based clustering, Bayesian clustering, and Fuzzy C-means clustering, which can be used as alternatives or supplements to the K-means algorithm. The selection of an appropriate clustering algorithm is crucial, considering the specific characteristics of the data and the objectives of the research, as each algorithm possesses distinct strengths and limitations.

5.3 Advanced-Pixel Path Planning Algorithm using K-means clustering (Fast Advanced-Pixel)

As seen in the previous chapter, reducing the dimensionality of the optimization problem is a demand to increase the performance of the Pixel strategy. As the Pixel strategy is based on the Travelling Salesman Problem (TSP), dividing the nodes into *K* clusters could effectively decrease computational time and enhance algorithm performance. This is because the time complexity of an exact TSP algorithm grows exponentially with the number of cities, meaning that the computation time becomes unfeasible for large datasets. By dividing the total number of nodes into clusters, the large dataset is transformed into smaller data clusters, making it easier to manage and reducing the computation time to solve each cluster. This results in a more efficient and faster solution to the problem.

In the Pixel strategy algorithm (and its variations), the main user input is the distance between nodes so that the layer surface can be completely discretized by them, as shown in Figure 5.2(a). After that, with the Fast Advanced-Pixel proposal as a new, the user has a second input (as is customary in the K-means algorithm), which is the number of clusters, *K*. With this, the algorithm divides the nodes into clusters and presents the user with this division, as shown in Figure 5.2(b). The number of clusters is applied to all layers (whether equal or not), as shown in Figure 5.2(c), and the user can navigate through them and verify the results. If a layer is not as desired, the user can only modify the number of clusters for that particular layer.

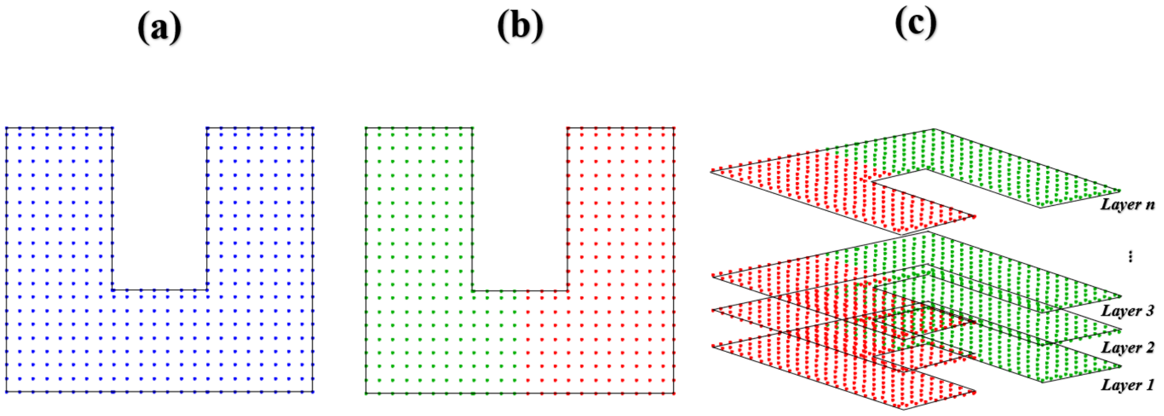


Figure 5.2 - Visualization of the Fast Advanced-Pixel strategy in Wire Arc Additive Manufacturing: (a) surface discretized by nodes with a user-defined distance between nodes; (b) clustering of layer surface nodes with user-defined number of clusters (K=2), shown as green and red clusters; (c) clustering applied to all layer surfaces

Depending on the topology of the part, the K-means algorithm may have difficulties in generating clusters with continuous nodes, as shown in Fig 5.3. Continuous nodes are defined here as nodes belonging to a cluster that will not be completely divided by islands (Fig 5.3(a)), entrances (Fig 5.3(b)) or holes (Fig 5.3(c)) in the layer geometry of the part (see Fig 5.3). Using Figure 5.3(a) as an example, it shows a non-continuous node division (or cluster), which is highlighted by blue squares. It can be seen the square hole at the centre of the layer divide the blue cluster into two well-defined sub-clusters. This is only an example, not meaning that this will occur with this specific layer shape. Figures 5.4(a) and 5.4(b), in turn, show continuous node divisions with K=2 and K=4 for the example demonstred in the Fig 5.3(c), respectively. A viable solution (though still manual) is the option to select another K value (an action taken by the user) so that clusters with continuous nodes are generated. It is important to emphasize that the presence of continuous nodes serves a crucial role in preventing trajectories from extending beyond the edges of the part. This is achieved by ensuring there are no gaps (the can be caused by holes, entrances and islands) among the nodes, thus facilitating the generation of trajectories that remain confined within the desired boundaries. It is worth noting that other techniques may be studied to generate clusters with continuous nodes in more complex geometries, which are, however, beyond the scope of this work.

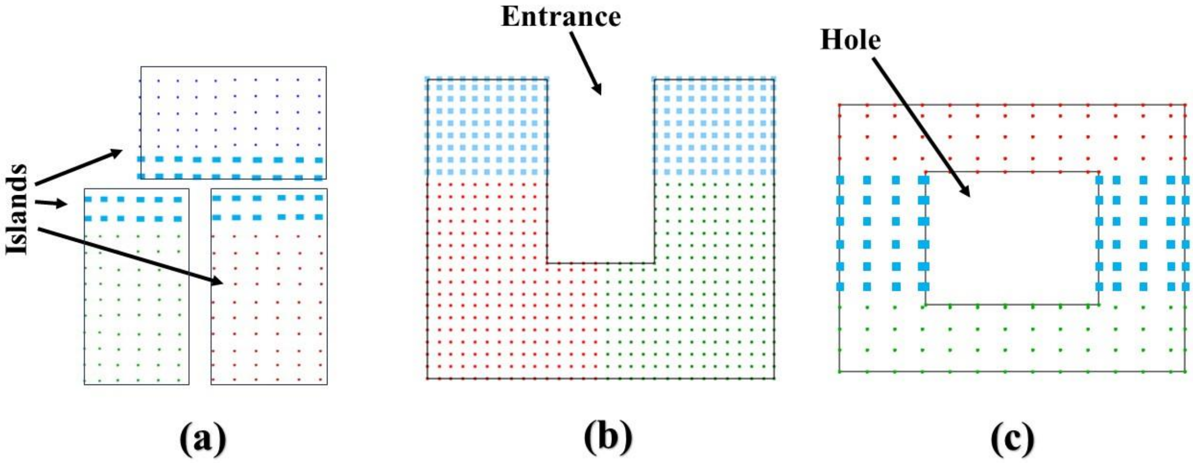


Figure 5.3- Examples of non-continuous node division, represented by blue squares, in geometries with: (a) island; (b) entrance; and (c) hole

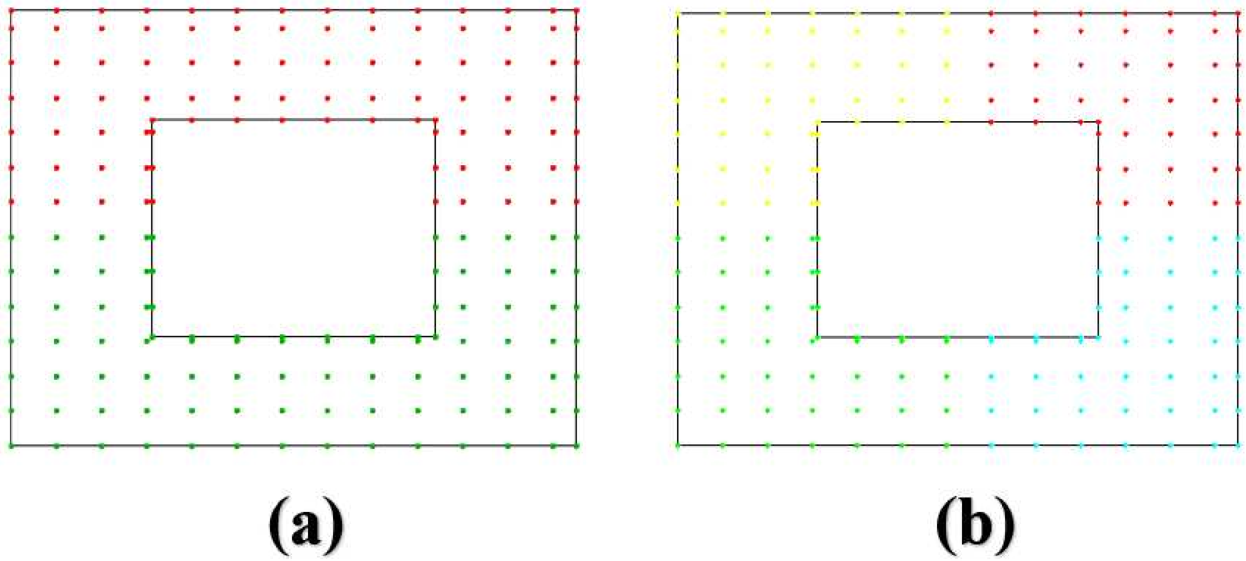


Figure 5.4 – Fast Advanced-Pixel strategy applied with different numbers of clusters (K): (a) Continuous node division, K=2; (b) Continuous node division, K=4

Figure 5.5 illustrates the potential to adjust the K value between layers. This possibility of generating different trajectories for each layer presents a promising strategy for mitigating issues such as lack of fusion, as highlighted by Wang et al. (2019). While acknowledging this possibility, it is important to note that this work does not explore this approach through experimental testing.

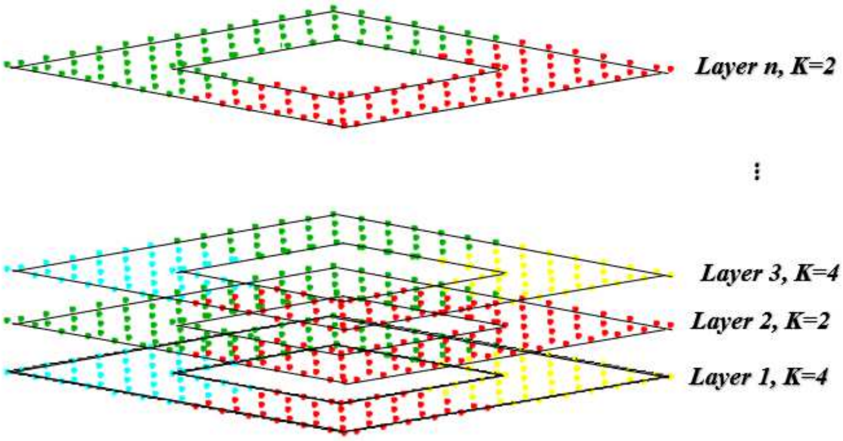


Figure 5.5 - Sliced layers with variable cluster numbers

With the defined clusters, trajectories can be generated in all clusters, independently, sequentially or in parallel (not explored in this work), using the Advanced-Pixel strategy. To clarify, the term "Fast Advanced-Pixel" refers to an accelerated version of the Advanced-Pixel method that incorporates a clustering technique. Although the core principles of the Advanced-Pixel approach are retained, the inclusion of the clustering technique enables significant computacional speed enhancements (as will be demonstrated in the upcoming sections). Remembering, Chapter IV explains that the user must define the policy and the number of iterations for the Advanced-Pixel algorithm to generate the trajectories.

After generating the trajectory for each cluster, it is necessary to establish interconnections between them. The user does this task. To demonstrate the connection between clusters and trajectories, Figure 5.6(a) schematise two clusters (represented in red and green) generated for a layer with a square shape. Then, Figure 5.6(b) shows the planned trajectory that could be hypothetically developed for the two clusters. As the next step, the trajectories generated for each cluster are joined together through a command from the user. This command is shown in detail in Figure 5.6(c), represented by a blue x and an arrow, and consists of selecting a point, referred to here as reference points, in the space between two consecutive clusters. With this reference point, the Fast Advanced-Pixel algorithm identifies the two points, for both clusters, as close to each other as possible. Figure 5.6(d) indicates that the nodes 1 and 2 (belonging to the green cluster and the closest ones to the reference point) and the closest nodes 3 and 4 (pertaining to the red cluster) are connected to each other in sequence. Reference point selection must occur until all clusters are interconnected (trajectory linking). Then, Figure 5.6(e) shows the final trajectory after joining the trajectories generated for each cluster, with node 1 forming a path with node 3, and node 2 forming a path with node 4. It is true that the selection of points can lead to human errors, but nothing complex to do. It is worth noting that the choice of reference point(s) does not necessarily have to be done in all layers, unless the layers have different topologies.

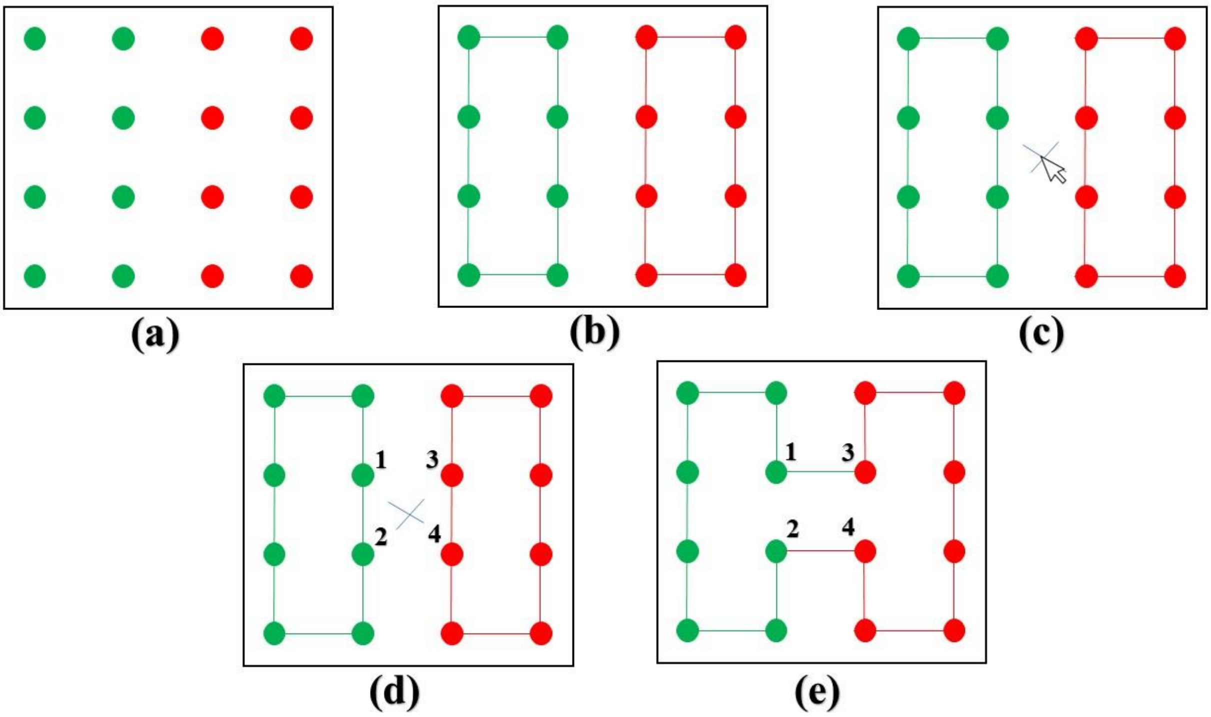
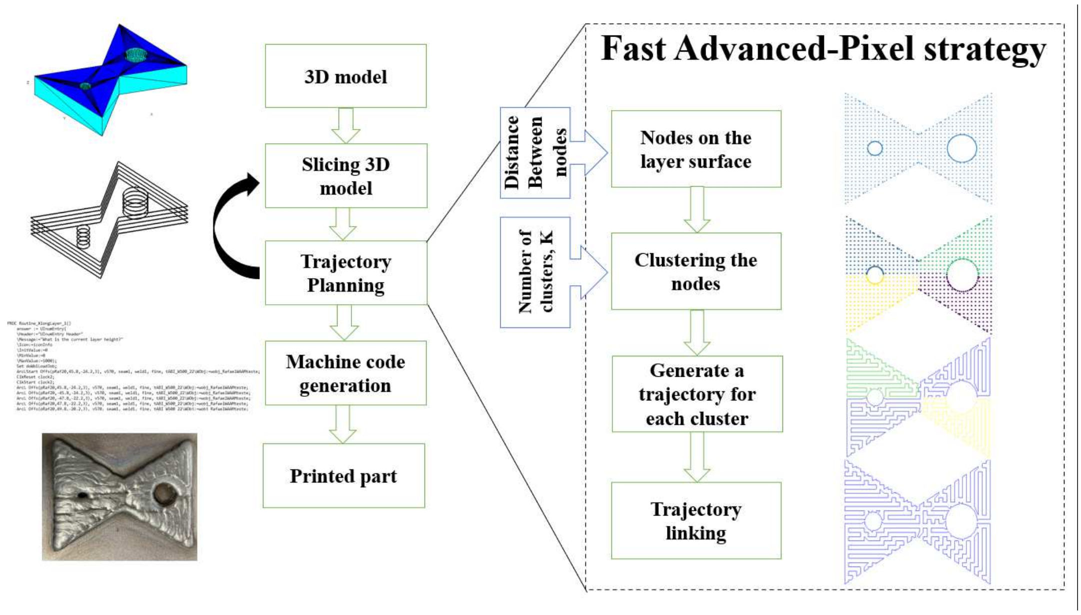


Figure 5.6 - Steps for trajectory generation using Fast Advanced-Pixel strategy: (a) clustering the nodes by K-means (implemented into the Fast-Advanced-Pixel); (b) generating trajectories for each cluster by Advanced-Pixel (integrated with the Fast_Advanced-Pixel); (c) selecting a reference point by the user (seen the white arrow); (d) identifying the nearest nodes (for each cluster) from the reference point (done by the Fast-Advanced Pixel); and (e) linking the trajectories based on the nearest points (done by the Fast-Advanced Pixel)

In summary, Figure 5.7 illustrates the 3D printing workflow, starting from the 3D model. The model is then sliced and prepared for trajectory planning through the Fast Advanced-Pixel strategy. In this strategy, the user inputs the distance between nodes (shown in the step “Nodes on the layer surface”) and the number of clusters, K (shown in the step “Clustering the nodes”). Clusters are generated for each layer (shown in the step “Generate a trajectory for each cluster”). The trajectories are, then, joined together using reference points selected by the user. This process is repeated until all clusters are connected, as shown in Fig. 5.6, resulting in the final trajectory (shown in the step “Trajectory linking”). Finally, the machine code is generated, and the part is printed.



```
PROC Routine_XiongLayer_1()
    answer := UINumEntry(
    \Header:="UINumEntry Header"
    \Message:="What is the current layer height?"
    \Icon:=iconInfo
    \InitValue:=0
    \MinValue:=0
    \MaxValue:=1000);
    Set doWblLoadInn;
    ArcLStart Offs(pRaf20,45.8,-24.2,3), v570, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ClkReset clock2;
    ClkStart clock2;
    ArcL Offs(pRaf20,45.8,-24.2,3), v570, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf20,-45.8,-24.2,3), v570, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf20,-47.8,-22.2,3), v570, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf20,47.8,-22.2,3), v570, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
    ArcL Offs(pRaf20,49.8,-20.2,3), v570, seam1, weld1, fine, tABI_W500_22\WObj:=wobj_RafaelWAAMteste;
```

Figure 5.7 - Overview of the 3D Printing workflow using Fast Advanced-Pixel Strategy

5.4 Computational Validation

This section aims at describing the assessment of whether the Fast Advanced-Pixel strategy offers any computational advantages compared to the Advanced-Pixel strategy, per objective SO 5.1. This section comprises the own methodology, presentation of results, and discussions. In order to make a fair comparison between the two strategies, the same simulation parameters will be used throughout the experiments. This study aims not to evaluate the efficacy of the path planning algorithms (this is the purpose of the next section), but rather to compare the computational performance of the two strategies. Specifically, the experiments will focus on the computational time and response quality of the trajectories generated by the two methods. These evaluations will be conducted using quantitative analysis.

5.4.1 Methodology

To achieve the objective stated in the previous section, four different parts (and their respective numbers of nodes) presented in Figure 5.8 were studied, using the following path-planning strategies:

1. Advanced-Pixel strategy, utilizing the Thompson Sampling tool policy (see Chapter IV), which was chosen arbitrarily due to no need for parameterization (this can be considered as a Fast Advanced-Pixel strategy with k = 1);
2. Fast Advanced-Pixel strategy with a value of k=2;
3. Fast Advanced-Pixel strategy with a value of k=4;
4. Fast Advanced-Pixel strategy with a value of k=8.

These strategies were selected to investigate the computational advantage of the Fast Advanced-Pixel strategies over the Advanced-Pixel strategy, with a particular focus on the effect of varying the value of *k* on computational time and response quality (trajectory distance). The experiments were conducted using a quantitative analysis approach.

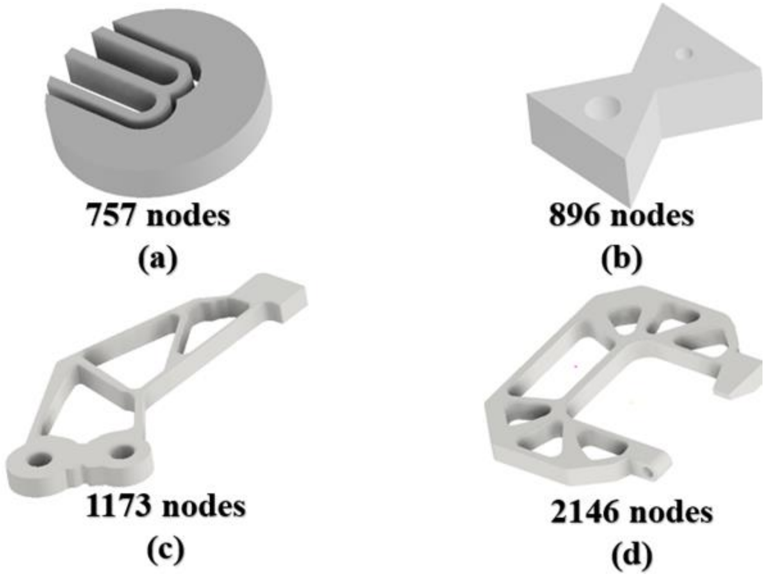


Figure 5.8- Tested parts for computational validation: (a) Binzel logo; (b) Tie-shaped part (Wang et al., 2019); (c) Jaw gripper with topology optimization; and (d) C-Frame of a resistance spot welding (RSW) gun with topology optimization (remembering, the number of nodes depends on the spacing and layer area)

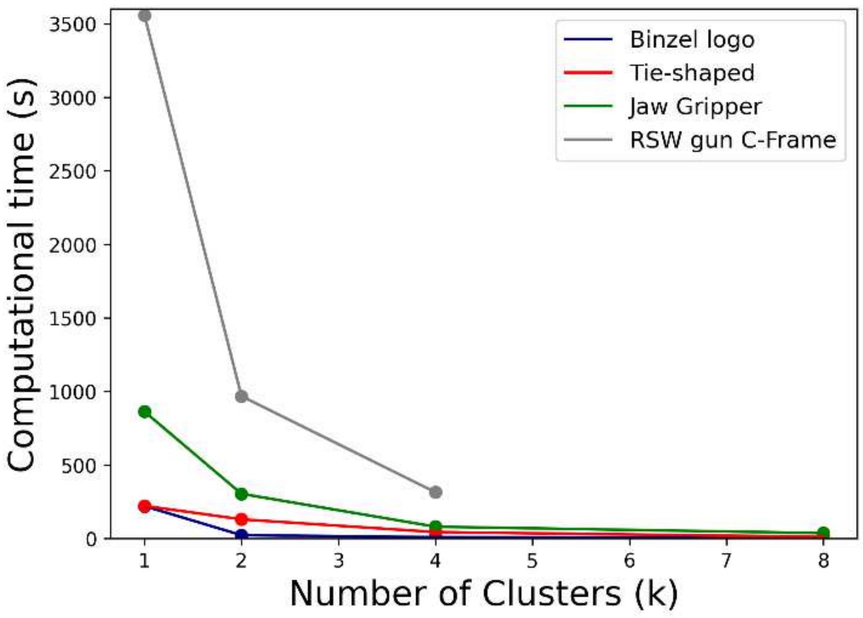
For each cluster *k*, a total of 100 iterations were performed on all the parts studied. For clusters with a value greater than 1, the user performed the operation of linking the trajectories (the time required for this manual operation was not included in the calculations). Due to the stochastic nature of the algorithm and the potential variation in results (between different *k* values), 5 independent runs were conducted for each experiment, and the distribution of the data was taken into consideration in the evaluation and discussion of the results. The experiments were conducted on a computer equipped with an Intel Core i7-12700H processor with 14 cores and 20 threads, operating at a frequency of 2.30 GHz and 16 GB of RAM (still with an Nvidia Titan V graphics card with 5120 CUDA cores and 12 GB of RAM). The criteria evaluated in this article were the computational time (in CPU time) for trajectory generation (response efficiency) and trajectory distance (response quality). To compare the outcomes achieved for different clusters assigned to the same part and draw conclusive findings regarding their significance, an analysis of variance (ANOVA) was employed at a confidence level of 0.95.

5.4.2 Results and Discussions

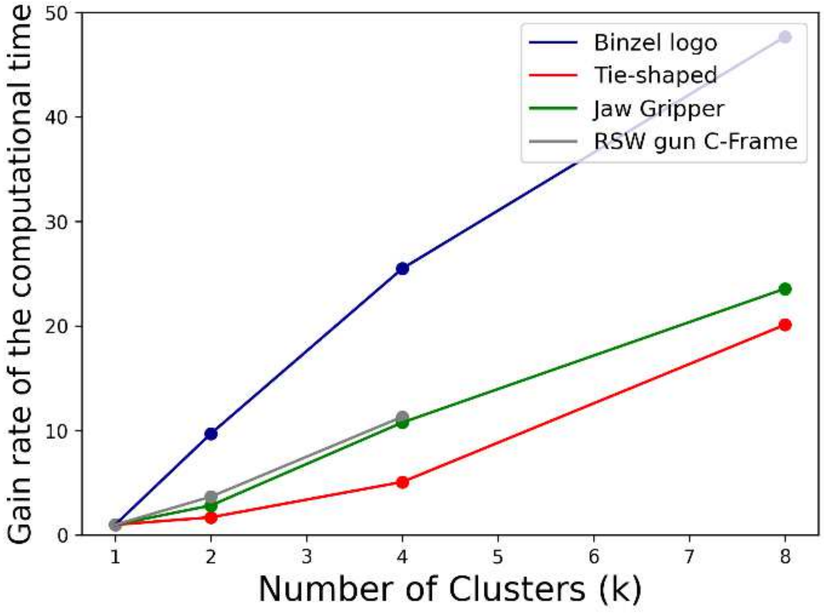
Table 5.1 shows the resulting trajectory distance and computational time for each value of *k* analysed in all studied parts. To facilitate data analysis, Figure 5.9(a) presents the computational time from all parts for each clustering value, while Figure 5.9(b) presents the gain rate of computational time in relation to cluster k=1. This latter value is obtained by dividing the computational time achieved for k=1 by the computational time for the other k values. It can be observed from the computational time results, according to Figure 5.9(a), that the time decreases in an exponential manner for all parts. This occurs because each value of *k* has a different number of nodes, as shown in Table 5.2. Therefore, for an NP-Hard problem like the travelling salesman problem, the computational complexity is O(n), meaning that complexity is a function of the number of initial points. When the number of initial points is reduced, the time is also reduced, and in the case of cluster division, the times are added serially (in this serial approach), thus reducing the total computational time. It is worth noting that the value of k = 8 were not employed when generated the clusters for RSW gun C-Frame due to the issue depicted in Figure 5.3(c). Therefore, the respective Computational time and Trajectory distance do not exist for this case study.In Figure 5.9(b), a trend of computational time gain rate can be observed in parts with a higher number of clusters (consequentely smaller clusters in size). Larger (in size) clusters demand longer processing times. Therefore, the reduction of the clusters size (by clustering) also decrease the processing time, making more significant the difference of the computational times. However, it is believed that very high cluster values may require further subdivision into clusters to exhibit this behaviour. In general, the standard deviation values tend to decrease as the number of clusters increases, since the smaller number of nodes induces less variability in the results due to the ease (faster) of converging to a local or even global optimum. Through ANOVA, the p-values were all below 0.05, indicating that the averages are different from each other. In this text, it is not important to highlight which division will be the best, but rather that division that can reduce computational time. Therefore, the gain rate of the computational time by increasing the number of clusters, reaching values up to 20 times faster when using 8 clusters.

Table 5.1 - Computational time, in seconds, and trajectory distance, in mm, for each part studied divided by *k* clusters

Criteria	*k* values	Parts			
		Binzel logo	Tie-shaped	Jaw Gripper	RSW gun C-Frame
Computational time (s)	1	210.31±5.54	221.75 ±8.67	862.96±30.00	3558.19±708.8
	2	22.68±3.24	130.09±9.34	303.90±13.96	967.78±29.62
	4	9.64±2.12	43.6±3.74	80.1±5.69	314.22±12.64
	8	4.62±1.10	11.02±1.47	36.60±4.28	
	p-value	3.05649E-20	2.08663E-18	5.5475902E-21	4.12339E-06
Trajectory distance (mm)	1	2496.76±10.24	2668.67±12.20	2924.64±8.33	6747.82±68.59
	2	2472.12±6.42	2642.24±7.51	2921.82±18.32	6610.94±15.23
	4	2470.36±3.16	2632.04±4.11	2904.41±10.44	6582.47±23.85
	8	2447.42±3.14	2638.57±5.36	2878.30±9.88	
	p-value	2.30598E-05	2.60807E-05	8.25229E-05	2.04609E-05



(a)



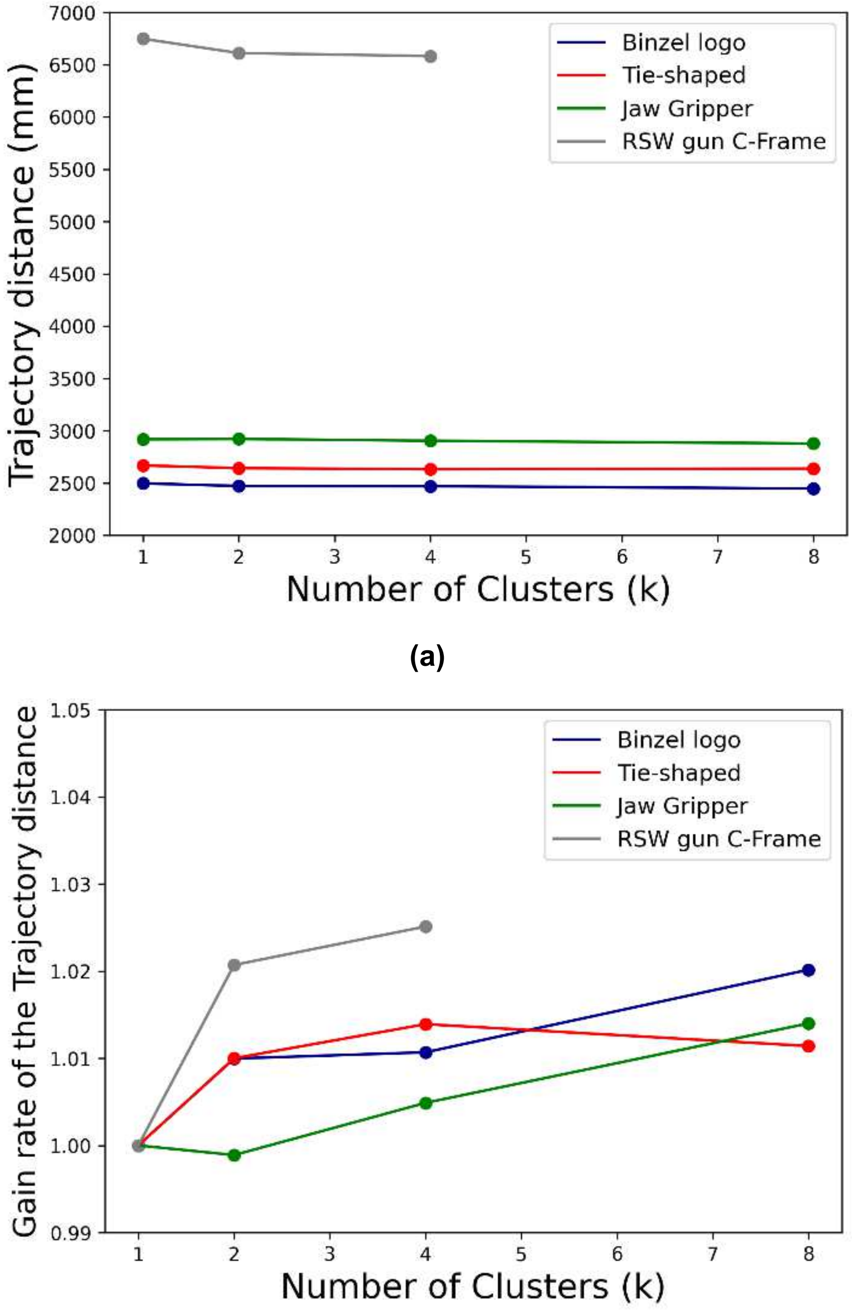
(b)

Figure 5.9 - Analysis of the studied parts related to k, with: (a) computational time; and (b) trend of grain rate of the computational time with respect to k=1

Table 5.2 - Number of nodes for the number of clusters in each part studied

Part	**Clusters (*k*)**														
	1	**2**		**4**				**8**							
	1	**1**	**2**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	757	381	376	170	171	206	210	106	70	102	109	106	82	97	85
2	896	458	438	221	221	233	221	107	81	129	108	90	132	118	131
3	1173	751	1173	306	290	312	265	108	93	242	173	121	99	192	145
4	2146	1108	1038	601	640	460	445								

Figure 5.10(a) displays the trajectory distance for all parts for each clustering value, while Figure 5.10(b) shows the gain rate of the trajectory distance concerning cluster k=1 (calculated in similar way from the gain rate of the computational time). From Figure 5.10(a), the trajectory distance may remain visually constant for all parts. However, the p-values added in Table 5.1 reveal significant differences in these values. It is important to note that this improvement is due to clustering, where having fewer nodes makes it easier to find the local or even global optimum in terms of trajectory distance. In Figure 5.10(b), the gain rate of the trajectory distance is high for the part with more nodes (RSW gun C-Frame), but the behaviour of the others does not follow a logical order.



(a)

(b)

Figure 5.10 - Analysis of the studied parts related to k, with: (a) trajectory distance and (b) trend of gain rate of trajectory distance with respect to k=1

It is important to note that number of nodes is not the only factor to consider here. Due to the different shapes (topologies) of the parts, they can also influence the results. However, it is important to emphasize that the objective of this study is to showcase the performance gain of the

Fast Advanced-Pixel strategy over the Advanced-Pixel strategy in different parts, rather than to demonstrate the influence of the number of nodes on the results for different values of *k*.

5.5 Case studies

5.5.1 Methodology

The parts depicted in Figures 5.8(c) and 5.8(d) were fabricated (WAAM), using the Fast Advanced-Pixel strategy. Their dimensions are shown in Figure 5.11. These parts were selected as functional demonstrators that had their topology optimized (as demonstrated in Figures 5.13 and 5.15). They exhibit geometrical complexities such as multiple intersections, different hole characteristics, and areas with varying thicknesses (evident in Figures 5.11(a) and (b), which contain slender regions with distinct thicknesses). Such features have been noticed as troublesome by various authors, as reported in the review by Jafari et al. (2021). Both the Jaw Gripper and the C-frame of the resistance spot welding (RSW) gun are examples of parts that can be fabricated via WAAM without requiring machining of their external surfaces, rendering it a more competitive process compared to other manufacturing methods. Only a few internal surfaces require machining, such as those that require hole trimming to accommodate the fastening elements.

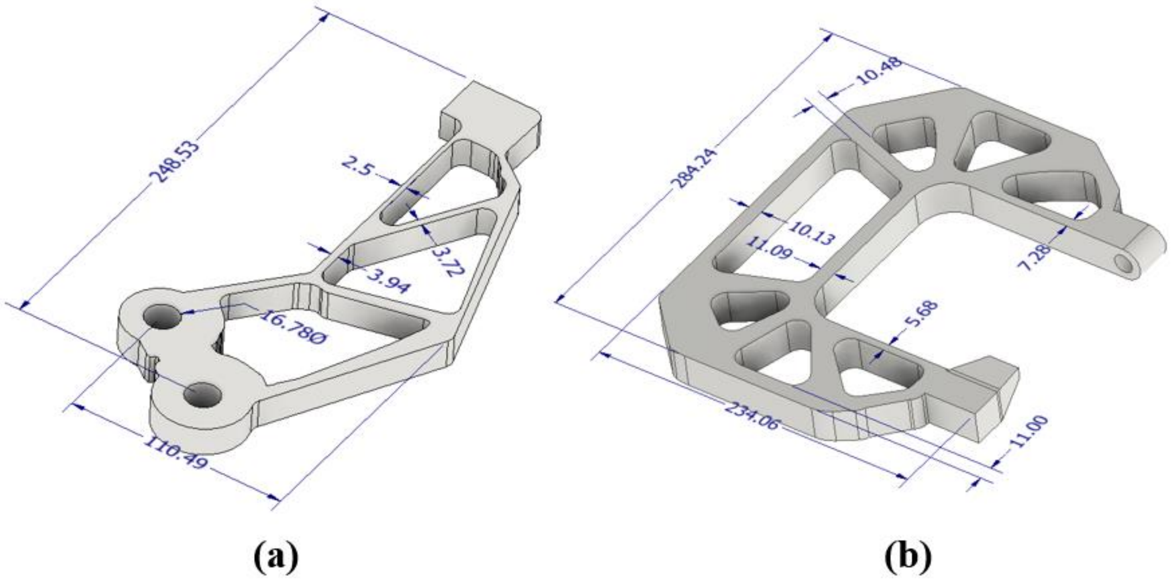


Figure 5.11 - Chosen parts as a case study and their respective dimensions: (a) Jaw Gripper; and (b) C-Frame of a Resistance Spot Welding Gun (dimensions in mm)

The parts were printed using Wire Arc Additive Manufacturing (WAAM) technology, following the workflow presented in Figure 5.12. The 3D models were uploaded, sliced, and the trajectories were generated by the version Fast Advanced-Pixel before the robotic code was produced. After generating the codes, they were loaded into the robot controller to begin printing. The deposition parameters, such as wire feeding speed and deposition speed, were all entered through the software's graphical interface and already passed to the robot controller through the generated code (see Table 5.3).

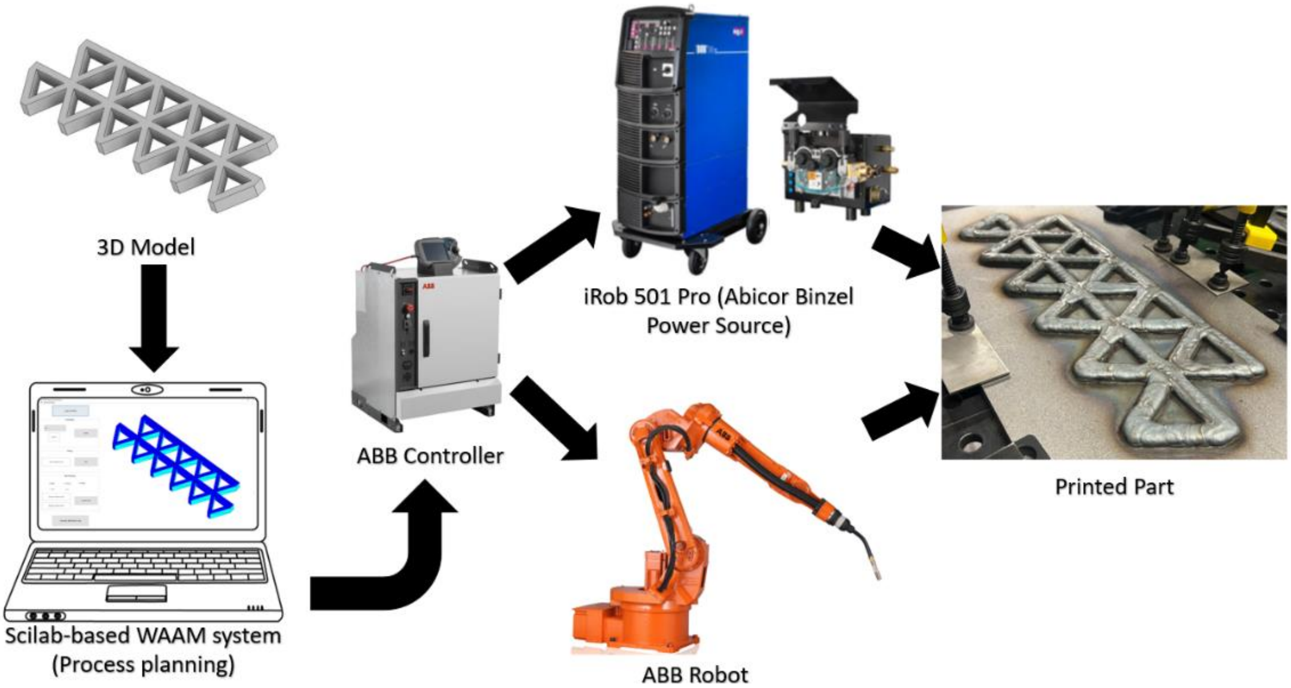

3D Model
Scilab-based WAAM system (Process planning)
ABB Controller
iRob 501 Pro (Abicor Binzel Power Source)
ABB Robot
Printed Part

Figure 5.12 - WAAM printing process chain of the parts used to validate the Fast Advanced-Pixel version experimentally

Table 5.3 - Experimental setting for printing the parts used to validate the Fast Advanced-Pixel version experimentally

Process	Kinects (Cold Metal Transfer technology from Abicor Binzel)
Arc deposition equipament	iRob 501 Pro
Torch movement system	ABB Robot IRB 1520 ID
Substrate	SAE 1020 carbon steel (300 mm x 300 mm x 12 mm)
Substrate cooling	air cooling
Wire (Material de adição)	AWS ER70S-6 - ϕ 1.2 mm
Shielding gas	Ar + 2%CO_2 - 15 L/min
Wire feed speed	3.7 m/min
CTWD*	12.0 mm
Deposition speed	48.0 cm/min
Corresponding Voltage	15.2 V
Corresponding Current	136 A

(*) CTWD – contact tip to work distance

The Fast Advanced-Pixel used the Thompson Sampling policy tool. A node spacing of 4 mm was selected based on the values of exploratory previously deposited beads-on-plate and following a lateral overlap ratio of 0.738, as used in similar conditions by Ding et al. (2015). Six layers of each part were printed per demonstrator. An interlayer temperature equal to or less than 80 °C was used as a reference. The values of k = 8, for the Jaw Gripper, and k = 4, for the C-Frame, of a resistance spot welding gun were selected for their satisfactory results in terms of computational time and response quality (see previous section). A total of 50 iterations were selected to generate the parts' trajectory. The two best results were used to print the odd and even layers, respectively. The software automatically chose a starting point and a printing direction before generating the robot codes for each layer.

After printing, the upper surface of the parts was machined by milling 5 mm deep to search for any internal non-conformity. The machined parts were compared to the 3D model using the 3-D scanner Twyn, a product of Visometry GmbH. This software employs augmented reality concepts to perform visual quality inspections. The printing times were measured using an internal command within the generated code for printing the parts.

5.5.2 Results

a) The Jaw Gripper

A simplified printing process of the Jaw Gripper is illustrated in Figure 5.13. To optimize the printing process using the Fast Advanced-Pixel strategy, the sliced-layer was divided into 8 clusters (as represented by cloured clusters in the figure), with the number of nodes listed in Table 5.2 Continuous and non-crossing trajectories were generated for printing the odd and even layers. Figure 5.14(a) shows the milling machined part with no visible non-conformities. The matching between the 3D CAD model and the printed piece is presented in Figure 5.14(b). It can be observed that the dimensions of the printed part are consistent with the 3D CAD model. The printing time was 446.77 and 445.69 seconds, respectively, for the odd and even layers) The part height achieved around 18 mm.

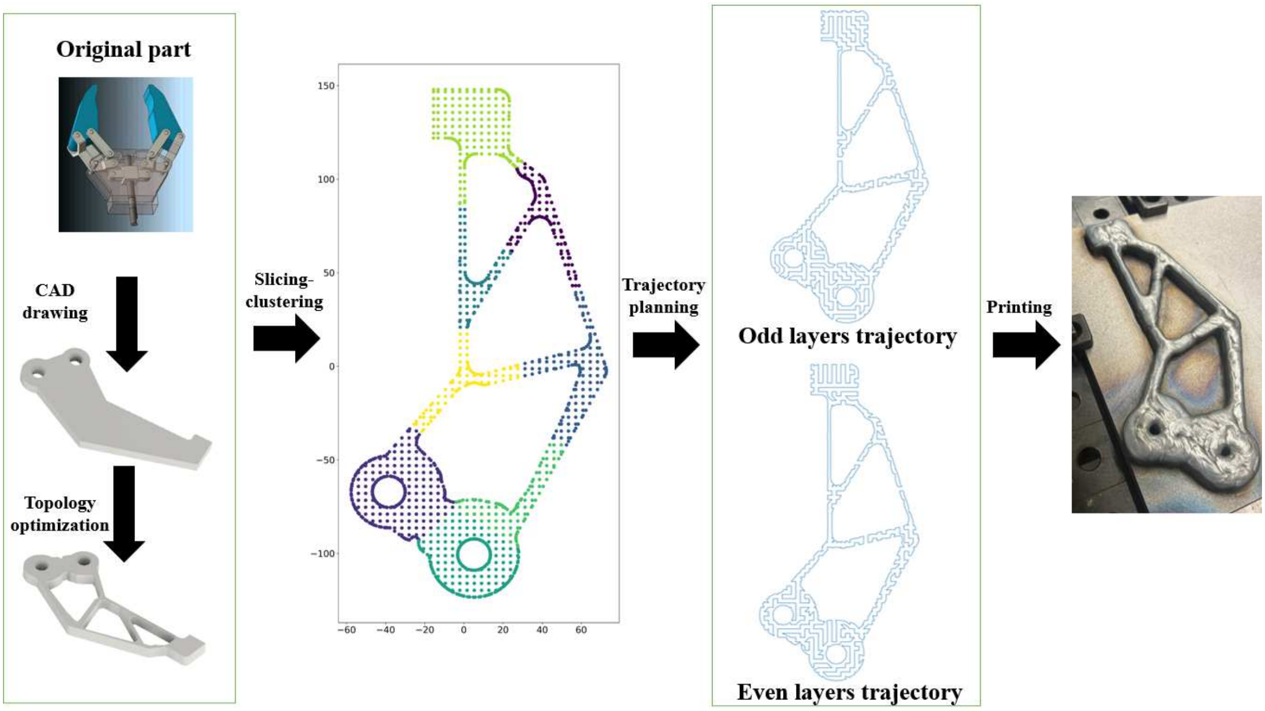


Figure 5.13 – Jaw Gripper: from the functional part to the printed part

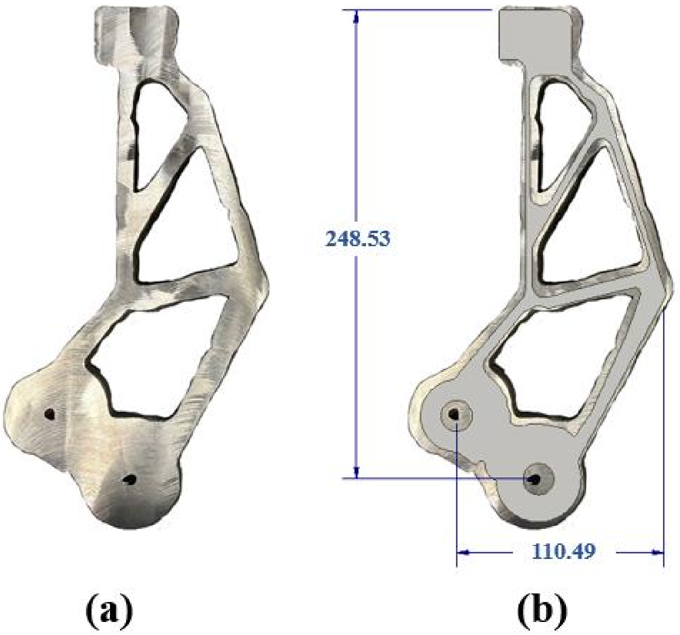


Figure 5.14 – Jaw Gripper: (a) Top view photo of the machined part; and (b) the 3D CAD model superimposed the machined part (dimensions in mm)

b) The C-Frame

Similarly to what was described for the Jaw Gripper, it is presented in Figure 5.15 the simplified printing process of the C-frame of the resistance spot welding gun. However, for this element, the part was divided into 4 clusters (shown in Figure 5.15 with the node divisions in different colours), with the number of nodes presented in Table 5.2. Figure 5.16(a) shows the machined surface of the part with no apparent discontinuities, and Figure 5.16(b) illustrates the matching between the 3D CAD model and the printed part. It is evident that the dimensions of the printed part match the 3D CAD model, even considering that only the first 6 layers were printed. The printing time was 1001 and 1004 seconds for the odd and even layers, respectively. The part height achieved around 18 mm.

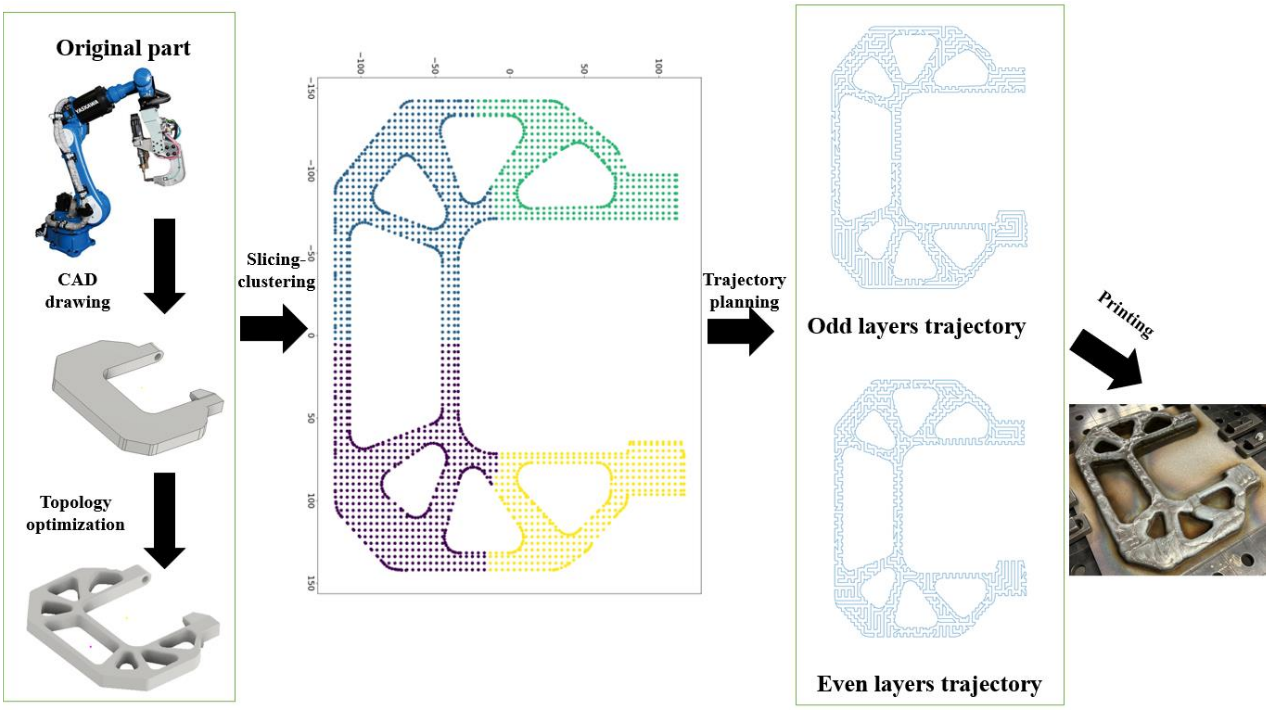


Figure 5.15 - C-Frame: from idea to the printed part

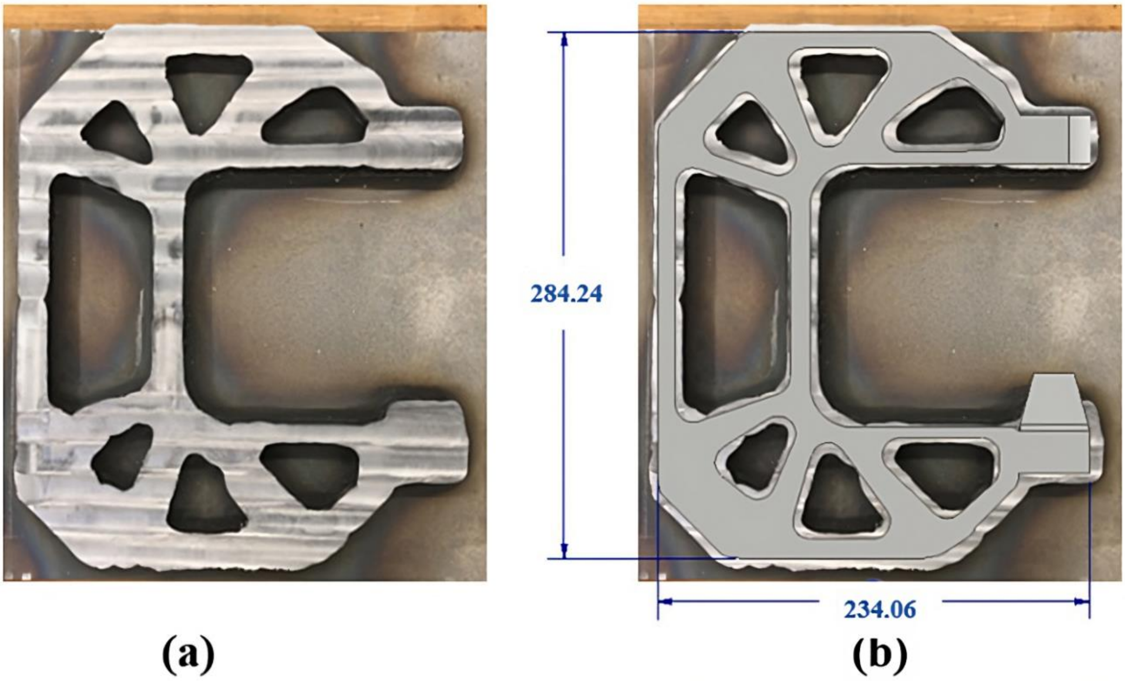


Figure 5.16 - C-frame of the resistance spot welding gun: (a) The machined part and (b) the 3D CAD model superimposed the machined part (dimensions in mm)

As described in objective SO 5.1, the case studies conducted on complex parts prove that the Fast Advanced-Pixel strategy effectively preserves the attained part quality discussed in the previous chapters. These studies demonstrate the absence of defects and the dimensional agreement between the 3D CAD model and the printed part. Although out of the scope of this work, a simple analysis using these present cases is shown in the Appendix, and it can be used to demonstrate how WAAM and using topology optimization can be profitable.

5.6 Partial conclusions

In conclusion, applying the K-means technique to boost the Advanced-Pixel (Fast Advanced-Pixel) strategy provided significant computational advantages over the standard Advanced-Pixel strategy for trajectory planning in 3D printing. By leveraging clustering techniques with k-means, the Fast Advanced-Pixel (which is the improvement of the Advanced-Pixel strategy, which in turn is a refinement of the Enhanced-Pixel strategy, started from the Basic-Pixel strategy) approach effectively reduces the dimensionality of the optimization problem. This reduction in complexity leads to notable improvements in computational time and objective function response quality (trajectory distance).

- Implementing the Fast Advanced-Pixel strategy enhanced performance for the Advanced-Pixel strategy algorithm across all tested parts. The computational time required for trajectory planning was significantly reduced, reaching values up to 20 times faster when using 8 clusters. In contrast, the overall response quality (gain rate of the trajectory distance) was notably improved, around 1 % shorter when using 8 clusters. This achievement aligns with the objective stated in SO 5.1.

- Furthermore, the successful application of the Fast Advanced-Pixel strategy was demonstrated by printing two complex parts. These printed components exhibit coherent dimensions when compared to the 3D model. Moreover, thorough inspections revealed no surface defects in printed parts or after post-printing machining. This outcome further demonstrated the effectiveness of the Fast Advanced-Pixel strategy. It maintains the quality achieved by the Advanced-Pixel, in meeting the requirements specified in SO 5.1.

CHAPTER VI

CONCLUSIONS

In conclusion, this thesis has successfully addressed the primary objective of investigating the potential benefits and challenges of implementing a Space-Filling strategy in Wire Arc Additive Manufacturing (WAAM). The development of the Pixel strategy has provided a flexible approach to trajectory planning for complex geometries, offering multiple applicable trajectories for part printing. Concerning the Basic-Pixel, the computational evaluation confirmed the effectiveness of the proposed novel Pixel strategy, with its four heuristics for node connections proving to be a suitable choice that can be optimized for different part geometries.

The experimental builds using Gas Metal Arc Welding (GMAW) and plain carbon steel demonstrated the practical viability of the Basic-Pixel strategy. It enabled the continuous deposition and construction of intricate shapes, including polygonal nonconvex geometries with holes, while maintaining the desired surface quality typical of WAAM.

The enhancements introduced by the Enhanced-Pixel strategy have positively impacted the performance of the Pixel algorithm. The new node sorting method and adding a new heuristic for trajectory planning have demonstrated improved trajectory distance and operational efficiency. Comparative analysis with conventional strategies such as Zigzag and Parallel Contour further confirmed the satisfactory operational efficiency and effectiveness of the Enhanced-Pixel strategy, particularly for the specific case studies conducted.

Overall, the Enhanced-Pixel strategy presented in this thesis holds promise for efficiently and effectively building complex parts. The successful application of the Multi-Armed Bandit (MAB) problem to the Enhanced-Pixel strategy (Advanced-Pixel) showcased the potential of reinforcement learning techniques for further enhancing performance.

Based on k-means clustering techniques, the Fast Advanced-Pixel strategy outperformed the standard Advanced-Pixel approach. It significantly reduced computational time, improved response quality, and demonstrated enhanced performance across the studied parts. Moreover, the strategy successfully printed complex parts with precise alignment to the 3D model and without surface defects even after post-printing machining. These outcomes emphasize the effectiveness of the Fast Advanced-Pixel strategy in optimizing trajectory planning and producing high-quality 3D printed parts.

As a whole, the research conducted in this thesis has advanced the field of WAAM trajectory planning by introducing the Pixel strategy. It can offer increased flexibility, operational efficiency, and effectiveness in printing complex geometries (potential benefits). However, challenges remain, such as the computational time increase with the size of the part and parts with non-constant cross sections. Nonetheless, advancements in hardware and software can address these challenges, and modifications to the Pixel algorithm, such as alternative axis ordering or alternative heuristics, can further improve its capabilities. Future research should focus on these areas to continue pushing the boundaries of WAAM trajectory planning.

Notwithstanding, one limitation of this study is that the experimental tests were conducted exclusively using carbon steel (although no difference is expected when the material is changed concerning trajectory planning, assuming that the parameters are input externally by the user).

CHAPTER VII

FUTURE WORKS

While this thesis has made significant progress in investigating and improving the Space-Filling strategy for Wire Arc Additive Manufacturing (WAAM), there are several avenues for future research and development in this field. The following areas are recommended for further exploration and exploitation:

1. Optimization of Computational Efficiency: the Space-Filling strategy, based on the Travelling Salesman Problem (TSP), faces challenges related to computational time, particularly for larger parts with non-constant cross-sections. Although advancements in hardware and software have improved the speed of solving the TSP, further research should focus on developing more efficient algorithms or techniques to minimize computational time without compromising the strategy's effectiveness. Exploring parallel computing, distributed computing, or utilizing specialized hardware could be potential directions to enhance computational efficiency;

2. Mitigation of Non-Conformities: the Pixel strategy, while effective in enabling the construction of complex geometries, may still exhibit non-conformities resulting from direction changes or other factors. Future research should investigate the development of closed-loop control systems (see item *b* below) or the incorporation of alternative heuristics, such as point distribution or ordering axes, to reduce or eliminate these non-conformities. Implementing suitable modifications to the Pixel algorithm based on empirical observations and computational simulations could lead to further improvements in the surface quality and dimensional accuracy of printed parts;

3. Exploration of Hybrid Strategies: while the Pixel strategy has demonstrated its flexibility and effectiveness in trajectory planning, there is potential for combining it with other strategies or algorithms to enhance its performance further. Future research could explore hybrid approaches that integrate the strengths of different trajectory planning techniques, such as

combining the Pixel strategy with other machine learning methods for adaptive trajectory optimization. These hybrid strategies have the potential to leverage the benefits of multiple approaches, leading to improved efficiency, accuracy, and surface quality in additive manufacturing processes;

Beyond the scope of this thesis, other suggestions for future work related to the subject:

a) Real-Time Monitoring and Quality Control: To ensure the quality and consistency of the printed parts, future research should focus on developing real-time monitoring and quality control systems. Integrating sensors and data analytics techniques can enable continuous monitoring of the printing process, detecting any deviations or anomalies and facilitating prompt adjustments or corrective actions. Real-time quality control can contribute to reducing waste, improving part quality, and increasing the overall efficiency of the additive manufacturing process;

b) Expansion to Different Materials: The experimental tests conducted in this study focused solely on carbon steel. To broaden the applicability of the Pixel strategy, future research should explore its performance with a wider range of materials commonly used in additive manufacturing, such as stainless steel, aluminium alloys, and titanium. Understanding how the strategy behaves with different material properties and their associated challenges will provide valuable insights for its implementation in real-world manufacturing scenarios.

In summary, the future works outlined above provide a roadmap for further advancements in the Space-Filling strategy for WAAM. Continued research in these areas will contribute to the broader adoption of additive manufacturing technologies, enabling the production of high-quality, complex parts with improved efficiency and reliability.

CHAPTER VIII

REFERENCES

ALJARRAH, O.; LI, J.; HUANG, W.; HERYUDONO, A.; BI, J. ARIMA-GMDH: a low-order integrated approach for predicting and optimising the additive manufacturing process parameters. **International Journal of Advanced Manufacturing Technology**, v. 106, p. 701-717, 2020. https://doi.org/10.1007/s00170-019-04315-8

ALMASRI, M.; MANSOUR, A.; MOY, C.; ASSOUM, A.; LE JEUNE, D.; OSSWALD, C. Distributed competitive decision making using multi-armed bandit algorithms. **Wireless Personal Communications**, v. 118, p. 1165-1188, 2021. https://doi.org/10.1007/s11277-020-08064-w

ALMEIDA, F. A.;GOMES, G. F.;DE PAULA, V. R.;CORRÊA, J. E.;DE PAIVA, A. P.;GOMES, J. H. F.;TURRIONI, J. B. A Weighted Mean Square Error Approach to the Robust Optimization of the Surface Roughness in an AISI 12L14 Free-Machining Steel-Turning Process. **Strojniški vestnik - Journal of Mechanical Engineering**, v. 64, n.3, p. 147-156, 2018. http://dx.doi.org/10.5545/sv-jme.2017.4901.

ISSN 0039-2480.AVŞAR, B.; ALIABADI, D. E. Parallelized neural network system for solving Euclidean traveling salesman problem. **Applied Soft Computing**, v. 34, p. 862-873, 2015. https://doi.org/10.1016/j.asoc.2015.06.011

BEKKER, A. C. M.; VERLINDEN, J. C. Life cycle assessment of wire + arc additive manufacturing compared to green sand casting and CNC milling in stainless steel. **Journal of Cleaner Production**, v. 177, 2018. https://doi.org/10.1016/j.jclepro.2017.12.148

BOUNEFFOUF, D.; RISH, I.; AGGARWAL, C. Survey on Applications of Multi-Armed and Contextual Bandits. **In 2020 IEEE Congress on Evolutionary Computation (CEC)**, p. 1-8, 2020. IEEE. https://doi.org/10.1109/CEC48606.2020.9185782

CATCHPOLE-SMITHA, S.; ABOULKHAIR, N.; PARRYA, L.; TUCKA, C.; ASHCROFT, I.A.; CLARE, A. Fractal scan strategies for selective laser melting of 'unweldable' nickel superalloys. **Additive Manufacturing**, v. 15, p. 113-122, 2017. https://doi.org/10.1016/j.addma.2017.02.002

CELEBI, M. E.; HASSAN A. K.; PATRICIO A. V. A comparative study of efficient initialization methods for the k-means clustering algorithm. **Expert Systems with Applications**, v. 40, n. 1, p.200-210, 2013. http://dx.doi.org/10.1016/j.eswa.2012.07.021

CHEN, G.; SHU, X.; LIU, J.; ZHANG, B.; FENG, J. A new coating method with potential for additive manufacturing: premelting electron beam-assisted freeform fabrication. **Additive Manufacturing**, v. 33, 2020. https://doi.org/10.1016/j.addma.2020.101118

COLOMO, A.G.; WOOD, D.; MARTINA, F.; WILLIAMS, S. A comparison framework to support the selection of the best additive manufacturing process for specific aerospace applications. **International Journal of Rapid Manufacturing**, v. 9, p. 194-211, 2020. https://doi.org/10.1504/IJRAPIDM.2020.107736

Consortium formed to further Wire Arc Additive Manufacturing in industry. METAL-AM, 2021. Available in: https://www.metal-am.com/consortium-formed-to-further-wire-arc-additive-manufacturing-in-industry/. Access in: 14/05/2021

COOKE, S.; AHMADI, K.; WILLERTH, S.; HERRING, R. Metal additive manufacturing: technology, metallurgy and modelling. **Journal of Manufacturing Processes**, v. 57, p. 978-1003, 2020. https://doi.org/10.1016/j.jmapro.2020.07.025

COX, J.J.; TAKEZAKI, Y.; FERGUSON, H.R.P.; KOHKONEN, K.E.; MULKAY, E.L. Space-filling curves in tool-path applications. **Computer-Aided Design**, v. 26, p. 215-224, 1994. https://doi.org/10.1016/0010-4485(94)90044-2

CUNNINGHAM, C. R.; FLYNN, J. M.; SHOKRANI, A.; DHOKIA, V.; NEWMAN, S. T. Invited review article: strategies and processes for high quality wire arc additive manufacturing. **Additive Manufacturing**, v. 22, p. 672-686, 2018. https://doi.org/10.1016/j.addma.2018.06.020

CUI, J.; YUAN, L.; COMMINS, P.; HE, F.; WANG, J.; PAN, Z. WAAM process for metal block structure parts based on mixed heat input. **The International Journal of Advanced Manufacturing Technology**, v. 113, p. 503–521, 2021. https://doi.org/10.1007/s00170-021-06654-x

DA SILVA, L. J.; SOUZA, D. M.; DE ARAÚJO, D. B.; REIS, R. P; SCOTTI, A.Concept and validation of an active cooling technique to mitigate heat accumulation in WAAM. **The International Journal of Advanced Manufacturing Technology**, v. 107, p. 2513–2523, 2020. https://doi.org/10.1007/s00170-020-05201-4

DEBROY, T.; WEI, H.L.; ZUBACK, J.S.; MUKHERJEE, T.; ELMER, J.W.; MILEWSKI, J.O.; BEESE, A.M.; WILSON-HEID, A.; DE, A.; ZHANG, W. Additive manufacturing of metallic components - Process, structure and properties. **Progress in Materials Science**, v. 92, p. 112–224, 2018. https://doi.org/10.1016/j.pmatsci.2017.12.001

Deutsche Bahn and Gefertec conclude pilot project for on-demand spare parts by AM. METAL-AM, 2019. Available in: https://www.metal-am.com/deutsche- bahn-and-gefertec-conclude-pilot-project-for-on-demand-spare-parts-by-am/. Access in: 14/05/2021

DING, D.; PAN, Z.; CUIURI, D.; LI, H. A tool-path generation strategy for wire and arc additive manufacturing. **International Journal of Advanced Manufacturing Technology**, v. 73, p. 173-183, 2014. https://doi.org/10.1007/s00170-014-5815-1

DING, D.; PAN, Z.; CUIURI, D.; LI, H. Wire-feed additive manufacturing of metal components: technologies, developments and future interests. **International Journal of Advanced Manufacturing Technology**, v. 81, p. 465-481, 2015a. https://doi.org/10.1007/s00170-015-7077-3

DING, D. Process planning for robotic wire ARC additive manufacturing. 2015. Doctoral thesis (Ph.D.) - School of Mechanical, Materials and Mechatronics Engineering, University of Wollongong, Wollongong, 2015b. Available at: http://ro.uow.edu.au/theses/4613

DING, D.; PAN, Z.; CUIURI, D.; LI, H. A practical path planning methodology for wire and arc additive manufacturing of thin-walled structures. **Robotics and Computer-Integrated Manufacturing**, v. 34, p. 8-19, 2015c. https://doi.org/10.1016/j.rcim.2015.01.003

DING, D.; PAN, Z.; CUIURI, D.; LI, H. A multi-bead overlapping model for robotic wire and arc additive manufacturing (WAAM). **Robotics and Computer-Integrated Manufacturing**, v. 31, p. 101–110, 2015d. https://doi.org/10.1016/j.rcim.2014.08.008

DING, D.; PAN, Z.; CUIURI, D.; LI, H.; LARKIN, N. Adaptive path planning for wire-feed additive manufacturing using medial axis transformation. **Journal of Cleaner Production**, v. 133, p. 942–952, 2016. https://doi.org/10.1016/j.jclepro.2016.06.036

DIRKX, R.; DIMITRAKOPOULOS, R. Optimizing Infill Drilling Decisions Using Multi-Armed Bandits: Application in a Long-Term, Multi-Element Stockpile. **Mathematical Geosciences**, v. 50, p.35–52, 2018. https://doi.org/10.1007/s11004-017-9695-9

DWIVEDI, R.; KOVACEVIC, R. Automated torch path planning using polygon subdivision for solid freeform fabrication based on welding. **Journal of Manufacturing Systems**, v. 23, p. 278–291, 2004. https://doi.org/10.1016/S0278-6125(04)80040-2

FERREIRA, R.P.; VILARINHO, L.O.; SCOTTI, A. Development and implementation of a software for wire arc additive manufacturing preprocessing planning: trajectory planning and machine code generation. **Welding in the World**, v. 66, p. 455–470, 2022. https://doi.org/10.1007/s40194-021-01233-w

FRÄNTI, P.; SIERANOJA, S. How much can k-means be improved by using better initialization and repeats?. **Pattern Recognition**, v. 93, p. 95-112, 2019. https://doi.org/10.1016/j.patcog.2019.04.014

GEBLER, M.; UITERKAMP, A. J. M. S.; VISSER, C. A global sustainability perspective on 3D printing technologies. **Energy Policy**, v. 74, 2014. https://doi.org/10.1016/j.enpol.2014.08.033

GERDJIJOV, S.; WOLFF, A. Decomposing a simple polygon into pseudo-triangles and convex polygons. **Computational Geometry**, v. 41, p. 21-30, 2008. https://doi.org/10.1016/j.comgeo.2007.10.005

GOMEZ, G.; CORTÉS, C.; CREUS, C.; AMILIBIA, M. Z.; MORENO, A. Generation of continuous hybrid zigzag and contour paths for 3D printing. **The International Journal of Advanced Manufacturing Technology**, v. 119, p. 7025–7040, 2022. https://doi.org/10.1007/s00170-021-08418-z

HAJAD, M.; TANGWARODOMNUKUN, V.; JATURANONDA, C.; DUMKUM, C. Laser cutting path optimization with minimum heat accumulation. **The International Journal of Advanced Manufacturing Technology**, v. 105, p. 2569–2579, 2019. https://doi.org/10.1007/s00170-019-04455-x

HU, Z.; QIN, X.; SHAO, T.; LIU, H. Understanding and overcoming of abnormity at start and end of the weld bead in additive manufacturing with GMAW. **The International Journal of Advanced Manufacturing Technology**, v. 95, p. 2357–2368, 2018. https://doi.org/10.1007/s00170-017-1392-9

HUTSEBAUT-BUYSSE, M.; METS, K.; LATRÉ, S. Hierarchical Reinforcement Learning: A Survey and Open Research Challenges. **Machine Learning and Knowledge Extraction,** v. 4, p.172-221, 2022. https://doi.org/10.3390/make4010009

ISO/ASTM 52900.2018, Additive manufacturing – general principles – terminology, international organization for standardization, Geneva, Switzerland, 2018. Available at: https://www.iso.org/obp/ui/#iso:std:74514:en. Accessed on: 01/05/2021

JACKSON, M. A.; VAN ASTEN, A.; MORROW, J. D.; MIN, S.; PFEFFERKORN, F. E. A comparison of energy consumption in wire-based and powder-based additive-subtractive manufacturing. **Procedia Manufacturing**, v. 5, p. 451-466, 2016. https://doi.org/10.1016/j.promfg.2016.08.087

JAFARI, D.; VANEKER, T. H. J.; GIBSON, I. Wire and arc additive manufacturing: opportunities and challenges to control the quality and accuracy of manufactured parts. **Materials & Design**, v. 202, p. 109471, 2021. https://doi.org/10.1016/j.matdes.2021.109471

JAIN, S.; BHAT, S.; GHALME, G.; PADMANABHAN, D.; NARAHARI, Y. Mechanisms with learning for stochastic multi-armed bandit problems. **Indian Journal of Pure and Applied Mathematics**, v. 47, p. 229–272, 2016. https://doi.org/10.1007/s13226-016-0186-3

JAIN, A. K. Data clustering: 50 years beyond K-means**. Pattern Recognition Letters**. v. 31, p. 651-666, 2010. https://doi.org/10.1016/j.patrec.2009.09.011

JORGE, V.L.; TEIXEIRA, F.R.; SCOTTI, A. Pyrometrical Interlayer Temperature Measurement in WAAM of Thin Wall: Strategies, Limitations and Functionality. **Metals**, v. 12, 2022. https://doi.org/10.3390/met12050765

KAPIL, S.; JOSHI, P.; YAGANI, H.V.; RANA, D.; KULKARNI, P.M.; KUMAR, R.; KARUNAKARAN, K.P. Optimal space filling for additive manufacturing. **Rapid Prototyping Journal**, v. 22, p. 660-675, 2016. https://doi.org/10.1108/RPJ-03-2015-0034

KOCH, P. H.; ROSENKRANZ, J. Sequential decision-making in mining and processing based on geometallurgical inputs. **Minerals Engineering**, v. 149, 2020. https://doi.org/10.1016/j.mineng.2020.106262

KUMAR, S.; GAUR, V.; WU, C. Machine learning for intelligent welding and manufacturing systems: research progress and perspective review. **The International Journal of Advanced Manufacturing Technology**, v. 123, p.3737–3765, 2022. https://doi.org/10.1007/s00170-022-10403-z

LAPORTE, G.; PALEKAR, U. Some applications of the clustered travelling salesman problem. **Journal of the Operational Research Society**, v. 53, p. 972-976, 2022. https://doi.org/10.1057/palgrave.jors.2601420

LEE, M.-T.; CHEN, B.-Y.; LAI, Y.-C. A Hybrid Tabu Search and 2-opt Path Programming for Mission Route Planning of Multiple Robots under Range Limitations. **Electronics** ,v. 9, 2020 https://doi.org/10.3390/electronics9030534

LI, Y., HAN, Q., ZHANG, G.; HORVÁTH, I. A layers-overlapping strategy for robotic wire and arc additive manufacturing of multi-layer multi-bead components with homogeneous layers. **The International Journal of Advanced Manufacturing Technology**, v. 96, p. 3331–3344, 2018. https://doi.org/10.1007/s00170-018-1786-3

LIN, S.; XIA, L.; MA, G.; ZHOU S.; XIE, Y.M. A maze-like path generation scheme for fused deposition modelling. **The International Journal of Advanced Manufacturing Technology**, v. 104, p 1509–1519, 2019. https://doi.org/10.1007/s00170-019-03986-7

LIU, H.H.; ZHAO, T.; LI, L.Y.; LIU, W.J.; WANG, T.Q.; YUE, J.F. A path planning and sharp corner correction strategy for wire and arc additive manufacturing of solid components with polygonal cross-sections. **The International Journal of Advanced Manufacturing Technology,** v. 106, p. 4879-4889, 2020. https://doi.org/10.1007/s00170-020-06071-1

MA, G.; ZHAO, G.; LI, Z.; YANG, M.; XIAO, W. Optimisation strategies for robotic additive and subtractive manufacturing of large and high thin-walled aluminum structures. **International Journal of Advanced Manufacturing Technology**, v. 101, p. 1275–1292, 2019. https://doi.org/10.1007/s00170-018-3009-3

MARKOVIĆ, D.; STOJIĆ, H.; SCHWÖBEL, S.; KIEBEL, S. J. An empirical evaluation of active inference in multi-armed bandits. **Neural Networks**, v. 144, p. 229-246, 2021. https://doi.org/10.1016/j.neunet.2021.08.018

MARTÍN, M.; JIMÉNEZ-MARTÍN, A.; MATEOS, A.; HERNÁNDEZ, J.Z. Improving A/B Testing on the Basis of Possibilistic Reward Methods: A Numerical Analysis. **Symmetry**, v. 13, 2021. https://doi.org/10.3390/sym13112175

MARTINA, F.; DING, J.; WILLIAMS, S.; CABALLERO, A.; PARDAL, G.; QUINTINO, L. Tandem metal inert gas process for high productivity wire arc additive manufacturing in stainless steel. **Additive Manufacturing**, v. 25, p. 545-550, 2019. https://doi.org/10.1016/j.addma.2018.11.022

MATOS, M. A.; ROCHA, A. M. A. C.; PEREIRA, A. I. Improving additive manufacturing performance by build orientation optimization**. International Journal of Advanced Manufacturing Technology**, v. 107, p. 1993-2005, 2020. https://doi.org/10.1007/s00170-020-04942-6

MICHEL, F.; LOCKETT, H.; DING, J.; MARTINA, F.; MARINELLI, G.; WILLIAMS, S. A modular path planning solution for Wire + Arc Additive Manufacturing. **Robotics and Computer-Integrated Manufacturing**, v. 60, p. 1-11, 2019. https://doi.org/10.1016/j.rcim.2019.05.009

MX3D's wire-arc additively manufactured industrial robot arm shortlisted in competition. METAL-AM, 2020. Available in: https://www.metal-am.com/mx3ds-wire-arc-additively-manufactured-industrial-robot-arm-short-listed-in-competition/. Access in: 14/05/2021

MU, H.; HE, F.; YUAN, L.; COMMINS, P.; WANG, H.; PAN, Z. Toward a smart wire arc additive manufacturing system: A review on current developments and a framework of digital twin. **Journal of Manufacturing Systems**, v. 67, p. 174-189, 2023. https://doi.org/10.1016/j.jmsy.2023.01.012

MUCA, M.; KUTROLLI, G.; KUTROLLI, M. A proposed algorithm for determining the optimal number of clusters. **European Scientific Journal**, v.11, 2015.

NEGI, S.; NAMBOLAN, A. A.; KAPIL, S.; JOSHI, P. S.; MANIVANNAN, R.; KARUNAKARAN, K. P.; BHARGAVA, P. Review on electron beam based additive manufacturing. **Rapid Prototype Journal**, v. 26, n. 3, p. 485-498, 2019. https://doi.org/10.1108/RPJ-07-2019-0182

NGUYEN, L.; BUHL, J.; BAMBACH, M. Continuous Eulerian tool path strategies for wire-arc additive manufacturing of rib-web structures with machine-learning-based adaptive void filling. **Additive Manufacturing**, v. 35, 2020. https://doi.org/10.1016/j.addma.2020.101265

OCHI, K.; KAMIURA, M. Overtaking method based on sand-sifter mechanism: Why do optimistic value functions find optimal solutions in multi-armed bandit problems? **Biosystems**, v. 135, p.55-65, 2015. https://doi.org/10.1016/j.biosystems.2015.06.009

Oil and gas industry consortium completes two projects to accelerate adoption of AM. 3D printing industry, 2020. Available in: https://3dprintingindustry.com/news/oil-and-gas-industry-consortium-completes- two-projects-to-accelerate-adoption-of-am-169056/. Access in: 14/05/2021

OUAARAB, A. Review of Tour Generation for Solving Traveling Salesman Problems. In: YANG, X. S.; ZHAO, Y. X. (eds) Nature-Inspired Computation in Navigation and Routing Problems. **Springer Tracts in Nature-Inspired Computing**. Springer, Singapore, 2020. https://doi.org/10.1007/978-981-15-1842-3_4

PAOLINI, A.; KOLLMANNSBERGER, S.; RANK, E. Additive manufacturing in construction: a review on processes, applications, and digital planning methods. **Additve Manufacturing**, v. 30, p. 100894, 2019. https://doi.org/10.1016/j.addma.2019.100894

RAMLAB unveils world's first class approved 3d printed ship's propeller. RAMLAB, 2017. Available in: https://www.ramlab.com/updates/ramlab-unveils-worlds-first-class-approved-3d-printed-ships-propeller/. Access in: 14/05/2021

RODRIGUES, T. A.; DUARTE, V.; MIRANDA, R. M.; SANTOS, T. G.; OLIVEIRA, J. P. Current status and perspectives on wire and arc additive manufacturing (WAAM). **Materials**, v. 12, p. 1121, 2019. https://doi.org/10.3390/ma12071121

SEBASTIAN, R.; CATCHPOLE-SMITH, S.; SIMONELLI, M.; RUSHWORTH, A.; CHEN, H.; CLARE, A. 'Unit cell' type scan strategies for powder bed fusion: The Hilbert fractal. **Additive Manufacturing**, v. 36, 2020. https://doi.org/10.1016/j.addma.2020.101588

SHI, C.; WEI, B.; WEI, S.; WANG, W.; LIU, H.; LIU, J. A quantitative discriminant method of elbow point for the optimal number of clusters in clustering algorithm. **EURASIP Journal on Wireless Communications and Networking**, v. 31, 2021. https://doi.org/10.1186/s13638-021-01910-w

SHI, J.; LI, F.; CHEN, S.; ZHAO, Y. T-GMAW based novel Multi-node trajectory planning for fabricating grid stiffened panels: An efficient production technology. **Journal of Cleaner Production**, v. 238, 2019. https://doi.org/10.1016/j.jclepro.2019.117919

SINGH, S.; SHARMA, S. K.; RATHOD, D. W. A review on process planning strategies and challenges of WAAM. **Materials Today: Proceedings**, v. 47, p. 6564-6575, 2021. https://doi.org/10.1016/j.matpr.2021.02.632

SINGH, S.; SINGH, A.; KAPIL, S.; DAS, M. Utilization of a TSP solver for generating non-retractable, direction favouring toolpath for additive manufacturing**. Additive Manufacturing**, v. 59, Part A, p. 103126, 2022. https://doi.org/10.1016/j.addma.2022.103126

SILVA, N.; WERNECK, H.; SILVA, T.; PEREIRA, A. C. M.; ROCHA, L. Multi-Armed Bandits in Recommendation Systems: A survey of the state-of-the-art and future directions. **Expert Systems with Applications**, v. 197, 2022. https://doi.org/10.1016/j.eswa.2022.116669

SOHRABI, S.; ZIARATI, K.; KESHTKARAN, M.A. Greedy randomized adaptive search procedure for the orienteering problem with hotel selection. **European Journal of Operational Research**, v. 283, p. 426–440, 2020. https://doi.org/10.1016/j.ejor.2019.11.010

SUBASI, A. **Clustering examples, Editor(s): Abdulhamit Subasi. Practical Machine Learning for Data Analysis Using Python**, Academic Press, chapter 7, p. 465-511, 2020. ISBN 9780128213797. https://doi.org/10.1016/B978-0-12-821379-7.00007-2

SUN, L.; REN, X.; HE, J.; ZHANG, Z. A bead sequence-driven deposition pattern evaluation criterion for lowering residual stresses in additive manufacturing. **Additive Manufacturing**, v. 48, Part A, p. 102424, 2021. https://doi.org/10.1016/j.addma.2021.102424

SUN, X.; MAZUR, M.; CHENG, C.T. A review of void reduction strategies in material extrusion-based additive manufacturing. **Additive Manufacturing**, v. 67, p. 103463, 2023,. https://doi.org/10.1016/j.addma.2023.103463

TAO, W.; LEU, M. C. Design of lattice structure for additive manufacturing. **International Symposium on Flexible Automation (ISFA)**, Cleveland, OH, USA, 2016, p. 325-332: https://doi.org/10.1109/ISFA.2016.7790182

TEIXEIRA, F.R.; SCOTTI, F.M.; VILARINHO, L.O.; DA MOTA, C. A. M.; SCOTTI, A. Transferability of the working envelope approach for parameter selection and optimisation in thin wall WAAM. **International Journal of Advanced Manufacturing Technology**, v. 119, p. 969–989, 2022. https://doi.org/10.1007/s00170-021-08326-2

TIBSHIRANI, R., WALTHER, G., HASTIE, T. Estimating the number of clusters in a data set via the gap statistic. **Journal of the Royal Statistical Society: Series B (Statistical Methodology)**, v. 63, p. 411-423, 2001. https://doi.org/10.1111/1467-9868.00293

VENTURINI, G.; MONTEVECCHI, F.; BANDINI, F.; SCIPPA, A.; CAMPATELLI, G. Feature based three axes computer aided manufacturing software for wire arc additive manufacturing dedicated to thin walled components. **Additive Manufacturing**, v. 22, p. 643-657, 2018. https://doi.org/10.1016/j.addma.2018.06.013

VISHWANATH, N.; SURYAKUMAR, S. Use of fractal curves for reducing spatial thermal gradients and distortion control. **Journal of Manufacturing Processes**, v. 81, p. 594-604, 2022. https://doi.org/10.1016/j.jmapro.2022.07.028

XIA, C.; PAN, Z.; POLDEN, J.; LI, H.; XU, Y.; CHEN, S.; ZHANG, Y. A review on wire arc additive manufacturing: Monitoring, control and a framework of automated system. **Journal of Manufacturing Systems**, v. 57, p. 31-45, 2020. https://doi.org/10.1016/j.jmsy.2020.08.008

XIE, H.; ZHANG, L.; LIM, C.P.; YU, Y.; LIU, C.; LIU, H.; WALTERS, J. Improving K-means clustering with enhanced Firefly Algorithms. **Applied Soft Computing**, v. 84, p. 105763, 2019. https://doi.org/10.1016/j.asoc.2019.105763

XIONG, J.; YIN, Z.; ZHANG, W. Forming appearance control of arc striking and extinguishing area in multi-layer single-pass GMAW-based additive manufacturing. **The International Journal of Advanced Manufacturing Technology**, v. 87, p. 579–586, 2016. https://doi.org/10.1007/s00170-016-8543-2

XIONG, Y.; PARK, S.; PADMANATHAN, S.; DHARMAWAN, A.G.; FOONG, S.; ROSEN, D.W.; SOH, G.S. Process planning for adaptive contour parallel toolpath in additive manufacturing with variable bead width. **The International Journal of Advanced Manufacturing Technology**, v. 105, p. 4159–4170, 2019. https://doi.org/10.1007/s00170-019-03954-1

WANG, X.; WANG, A.; LI, Y. A sequential path-planning methodology for wire and arc additive manufacturing based on a water-pouring rule. **The International Journal of Advanced Manufacturing Technology**, v. 103, p. 3813–3830, 2019. DOI: https://doi.org/10.1007/s00170-019-03634-2

WASSER, T., JAYAL, A.D., PISTOR, C. Implementation and evaluation of novel build styles in fused deposition modeling (FDM). **Strain** , v. 5, p. 95–102, 1999. http://dx.doi.org/10.26153/tsw/722

WIPPERMANN, A.; GUTOWSKI, T. G.; DENKENA, B.; DITTRICH, M.-A.; WESSARGES, Y. Electrical energy and material efficiency analysis of machining, additive and hybrid manufacturing. **Journal of Cleaner Production**, v. 251, p. 119731, 2020. https://doi.org/10.1016/j.jclepro.2019.119731

WU, Z.; SONG, T.; ZHANG, Y. Quantum k-means algorithm based on Manhattan distance. **Quantum Information Processing**, v. 21, 2022. https://doi.org/10.1007/s11128-021-03384-7

YUAN, L.; PAN, Z.; DING, D.; HE, F.; VAN DUIN, S.; LI, H.; LI, W. Investigation of humping phenomenon for the multi-directional robotic wire and arc additive manufacturing. **Robotics and Computer-Integrated Manufacturing**, v. 63, p. 101916, 2020. https://doi.org/10.1016/j.rcim.2019.101916

YANG, X.S. **Nature-Inspired Metaheuristic Algorithms**, 2nd ed.; Luniver Press: Cambridge, UK, 2008.

ZHANG, Y.; CHEN, Y.; LI, P; MALE, A. Weld deposition-based rapid prototyping: a preliminary study. **Journal of Materials Processing Technology**, v. 135, p. 347-357, 2003. https://doi.org/10.1016/S0924-0136(02)00867-1

ZIA, M.; CAKIR, Z.; SEKER, D.Z. Spatial Transformation of Equality – Generalized Travelling Salesman Problem to Travelling Salesman Problem. **ISPRS International Journal of Geo-Information**, v. 7, 2018. https://doi.org/10.3390/ijgi7030115

ZHAO, W-L.; DENG, C-H.; NGO, C-W. k-means: A revisit. **Neurocomputing**, v. 291, p. 195-206, 2018. https://doi.org/10.1016/j.neucom.2018.02.072

APPENDIX

An exploratory analysis of optimized topology parts printed through conventional strategies using WAAM

Although out of the scope of this work, a simple analysis using the studies cases shown in Chapter V can be used to demonstrate how WAAM when using topology optimization can be profitable. Additionally, two conventional strategies, namely Parallel-Contour and Zigzag, are computationally tested and analyzed.

Both printed parts shown in Chapter V (section 5.5) with topological can be compared with the manufacturing by machining in terms of material usage. Machined parts require high usage of material. For example, to manufacture the Jaw Gripper by milling, a solid metal block of at least 494.28 cm³ (248.53 x 110.49 x 18.00 mm) is needed, based on the maximum dimensions of the 3D model and the height of the printed part (18 mm). It is important to state that, commercially speaking, to find a block with this tied geometry is unlikely. In contrast, based on the wire diameter, wire feeding speed, and printing time for the 6 layers, the estimated material usage for printing the part was 186.95 cm³. For the C-frame spot welding gun, at least a solid metal block of 1197.52 cm³ (284.24 x 234.06 x 18.00 mm) of material is required for machining. However, based on the wire diameter, wire feeding speed, and printing time for the 6 layers, the estimated material usage for printing the part was 420.13 cm³. Therefore, the usage of material using WAAM technologies is 37% and 35% less than using machining, for the Jaw Gripper and C-frame spot welding gun, respectively.

Concerning other potential printing strategies using WAAM, Figures A1 and A2 illustrate possible problems that could arise. In Figure A1, with a focus on the Jaw Gripper, it is noted that both the Zigzag and Parallel Contour strategies (the blue lines represent the trajectories) may lead to geometric nonconformities. With the Parallel Contour strategy, unfilled areas (highlighted in red) may appear in the intersection regions (indicated in the green detail enlarged in Figure A1). Even when applying two parallel lines, unfilled areas may still occur. With the Zigzag strategy (considering a 90-degree rotation between layers), for the selected stepover distance between two parallels tracks of 4 mm, there is a possibility that a bar may not be printed in the right-to-left direction (in the case of 180 degrees), potentially compromising the entire printing process.

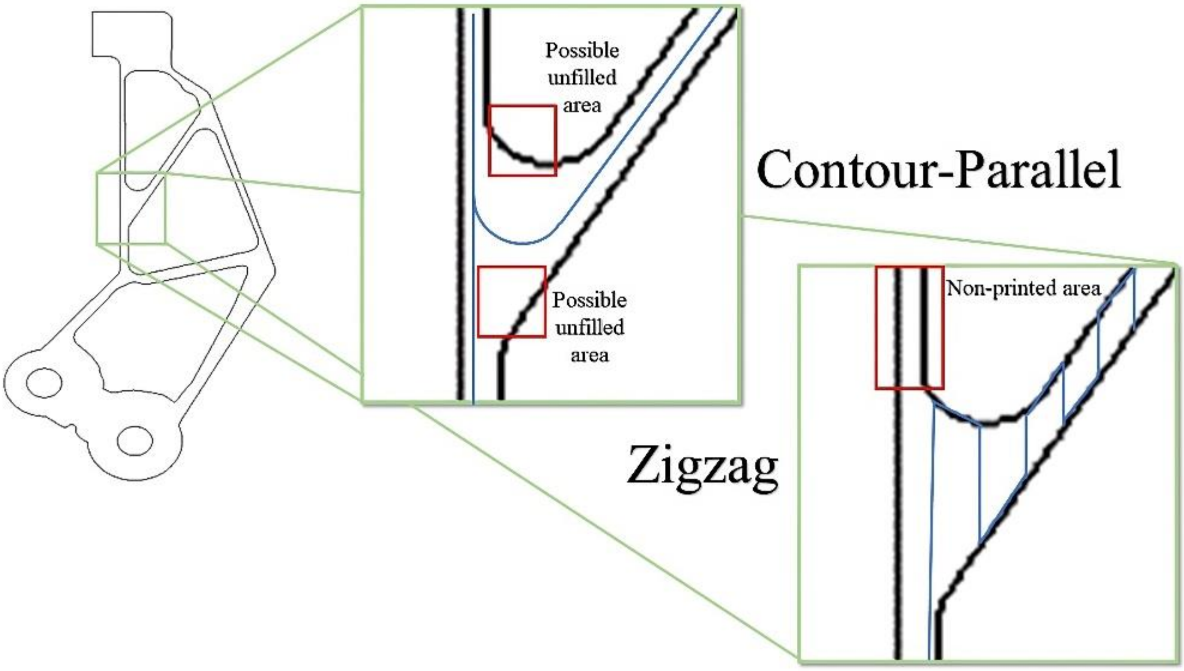


Figure A1 - Highlighted problems when printing the Jaw Gripper using conventional strategies

Refocusing on the C-frame resistance spot welding gun, Figure A2 shows that the Zigzag and Parallel Contour strategies can result in geometric nonconformities. The same intersection issues, highlighted by red as unfilled areas, arise when using the Parallel Contour strategy. Nevertheless, there is a risk of lines or regions being left unfilled due to the lack of a scan line (scan vector).

It is worth noting that conventional strategies (Zigzag and Parallel Contour) would generate a large number of movements without deposition for the printed parts, which can result in geometric non-conformities. Additionally, risks of failure to re-ignite the arc after an interruption during printing may occur. The latter possibility can compromise the printing process if the operator does not have a good knowledge of the machine code and the motion system used. Therefore, continuous trajectories are always welcome. Continuous trajectories are achieved by the Advanced-Pixel strategy at the cost of trajectories with a higher number of direction changes. These changes in direction can also result in geometric nonconformities, but can be resolved by applying different trajectories in alternate layers, as shown in Chapter III. However, it should be noted that a more advanced study aiming to reduce geometric nonconformities can be conducted using deep learning techniques to control the size of the molten pool during the printing, as related by Xia et al. (2020) and Mu et al. (2023).

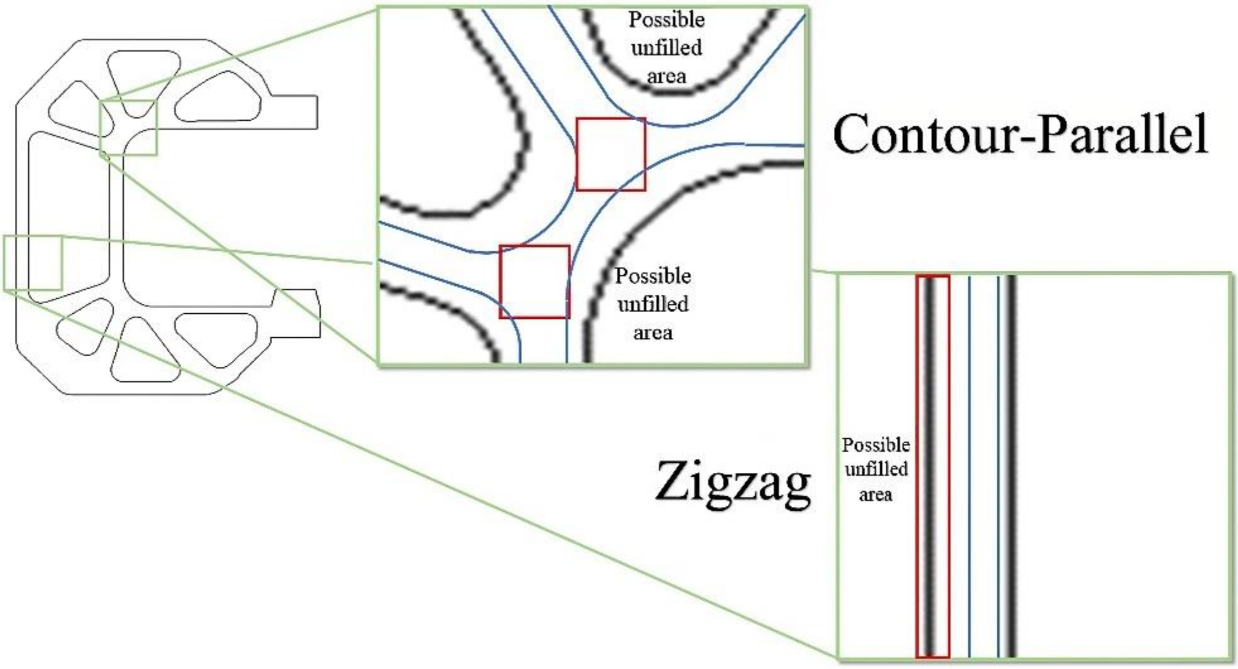


Figure A2 - Highlighted problems when printing the C-frame spot welding gun using conventional strategies