



UNIVERSITÀ
DEGLI STUDI
FIRENZE

FLORE

Repository istituzionale dell'Università degli Studi di Firenze

Feature based three axes computer aided manufacturing software for wire arc additive manufacturing dedicated to thin walled components

Questa è la Versione finale referata (Post print/Accepted manuscript) della seguente pubblicazione:

Original Citation:

Feature based three axes computer aided manufacturing software for wire arc additive manufacturing dedicated to thin walled components / Venturini, Giuseppe*; Montevicchi, Filippo; Bandini, Francesco; Scippa, Antonio; Campatelli, Gianni. - In: ADDITIVE MANUFACTURING. - ISSN 2214-8604. - ELETTRONICO. - 22:(2018), pp. 643-657. [10.1016/j.addma.2018.06.013]

Availability:

The webpage <https://hdl.handle.net/2158/1131519> of the repository was last updated on 2021-03-31T00:58:50Z

Published version:

DOI: 10.1016/j.addma.2018.06.013

Terms of use:

Open Access

La pubblicazione è resa disponibile sotto le norme e i termini della licenza di deposito, secondo quanto stabilito dalla Policy per l'accesso aperto dell'Università degli Studi di Firenze (<https://www.sba.unifi.it/upload/policy-oa-2016-1.pdf>)

Publisher copyright claim:

La data sopra indicata si riferisce all'ultimo aggiornamento della scheda del Repository FloRe - The above-mentioned date refers to the last update of the record in the Institutional Repository FloRe

(Article begins on next page)

Accepted Manuscript

Title: Feature based three axes computer aided manufacturing software for wire T arc additive manufacturing dedicated to thin walled components

Authors: Giuseppe Venturini, Filippo Montevecchi, Francesco Bandini, Antonio Scippa, Gianni Campatelli



DOI: <https://doi.org/10.1016/j.addma.2018.06.013>
Reference: ADDMA_427

To appear in:

Received date: 14/03/2018
Revised date: 28/05/2018
Accepted date: 17/06/2018

Please cite this article as: Giuseppe Venturini, Filippo Montevecchi, Francesco Bandini, Antonio Scippa, Gianni Campatelli, Feature based three axes computer aided manufacturing software for wire T arc additive manufacturing dedicated to thin walled components (2018), <https://doi.org/10.1016/j.addma.2018.06.013>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

Title

Feature based three axes computer aided manufacturing software for wire arc additive manufacturing dedicated to thin walled components

Authors

Giuseppe Venturini^{a*}, Filippo Montevercchi^a, Francesco Bandini^a, Antonio Scippa^a, Gianni Campatelli^a

^a *Affiliation:* Department of Industrial Engineering, University of Firenze, Via di Santa Marta 3, Firenze, 50139, Italy.

**Corresponding author.* Tel.: +39-055-2758726. E-mail address: giuseppe.venturini@unifi.it

Abstract

WAAM (Wire-Arc-Additive-Manufacturing) is a metal additive manufacturing process using arc welding to create large components with high deposition rate. The workpiece quality and the process productivity are strongly dependent both on the process parameters (wire feed speed, voltage and current) and on the selected deposition path. Currently, the CAM (Computer-Aided-Manufacturing) software dedicated to WAAM rely on a multi-pass strategy to create the component layers, i.e. each layer is built overlapping multiple welding passes. However, since WAAM can create wide layers, a single pass strategy can improve the process efficiency when dealing with thin walled components. This paper proposes CAM software dedicated to WAAM, using a single pass strategy. The proposed solution uses a midsurface representation of the workpiece as input, to generate the deposition toolpath. A feature recognition module is proposed, to identify the critical features of the part, such as free end walls, t-crossings, direct-crossings and isolated tubulars. A specific strategy is developed and proposed for each one of the selected features, with the aim of minimizing the geometrical errors and to ensure the required machining allowances for the subsequent finishing operations. The effectiveness of the proposed strategy is verified manufacturing a test case.

Keywords: Wire-Arc-Additive-Manufacturing, Gas-Metal-Arc-Welding, Computer-aided-manufacturing, Feature recognition.

1. INTRODUCTION

WAAM (Wire-Arc-Additive Manufacturing) is an AM (Additive Manufacturing) process using electric arc welding to create functional metal components. Compared to other AM technologies, WAAM is well suited to manufacture large components with relatively low features complexity [1,2]. This is an intrinsic feature of the arc welding processes (e.g. GTAW (Gas-Tungsten-Arc Welding) [3] or GMAW (Gas- Metal-Arc Welding) [4]) used in WAAM, which produce a wider bead than laser based direct energy deposition technologies which achievable bead width is limited by the laser beam spot size. WAAM can hence be regarded as a promising process to reduce the production cost and lead time of large, thin walled aerospace components, such as landing gear supports, flaps and wing ribs [5]. As all metal AM processes, WAAM requires dedicated software to automatically generate the deposition path to speed up the process planning phase. These algorithms perform a slicing operation, extracting the 2D geometry of the cross sections and then, using this data, define a suitable path to depose them. The algorithms to create the 2D section have been deeply investigated by several authors. Ding et al. [6] proposed to split the cross section in convex polygonal regions according to a divide-and-conquer strategy, depositing the regions with a combined contour and zigzag approaches, two common strategies in polymer AM. The same authors proposed an alternative strategy [7] based on the medial axis transformation, introduced by Kao and Printz [8], which avoids the formation of empty gaps inside the workpiece sections. Nilsiam et al. [9] modified an existing open source slicer (CORA) dedicated to FDM (Fused-Deposition-Modelling), to make it suitable for WAAM path generation.

All the presented algorithms use a multiple pass strategy to create the 2D sections, i.e. despite the thickness of the component wall, the deposition path consists of several parallel overlapped beads

[10]. This may not be convenient in the case of thin walled components. This paper considers a wall thin when it can be manufactured using a collection of stacked single beads, for a given welding apparatus. Fig. 1 clarifies the distinction between single and multiple passes strategies. Analyzing the existing literature, it can be stated that none of the proposed solutions explores the possibility of automatically creating the deposition toolpath for the component using a single welding pass for each layer.

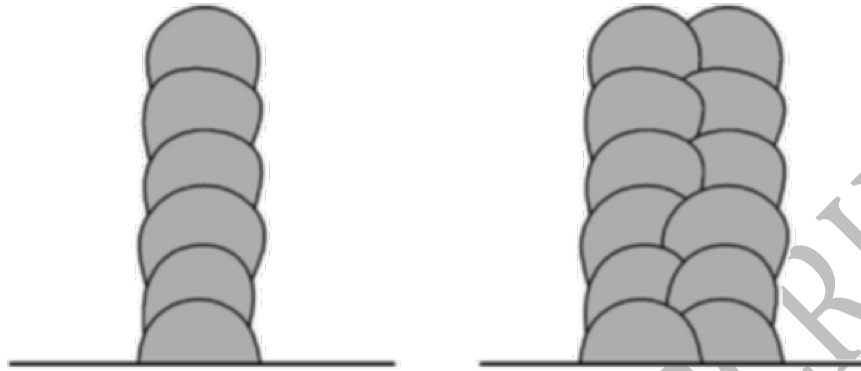


Fig. 1. A schematic cross section of a single bead wall (left) and an overlapped bead wall (right).

The single pass strategy exploits the possibility of adjusting the layer dimension by tuning the process parameters [11]. This is a convenient alternative for thin walled parts, since WAAM allows to deposit wide layers, achieving the required section geometry without bead overlapping, hence with a significant decrease of the manufacturing time. Moreover, bead overlapping requires the usage of specific models [12] to determine the overall width of overlapped beads, including further uncertainties in the path generation algorithms. This paper addresses this lack of the state of the art, proposing the base architecture of automated 3 axis CAM software for WAAM deposition without bead overlapping.

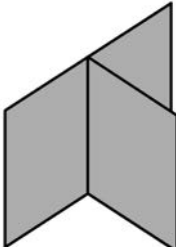
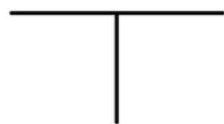
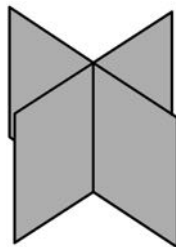
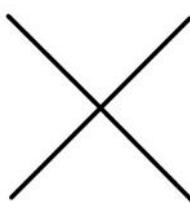
The most critical aspect of the single pass approach is represented by the crossing structures since an improper deposition strategy could lead to a nonuniform layer thickness. Moreover, different types of crossings require different strategies, as shown by Mehnen et al. [13] and Venturini et al. [14]. This paper proposes to overcome this issue by implementing a feature recognition algorithm, an approach successfully applied to different technologies, such as injection molding [15] and milling [16]. The internal workflow of the proposed CAM software is the following: the skeleton surface (midsurface) of the workpiece is imported in the CAM software; a feature recognition algorithm identifies the different crossings based on a proposed taxonomy; the deposition path is then generated by using specific strategies for each type of the detected features.

The following sections depict the proposed CAM, detailing: i) its overall architecture ii) the feature taxonomy and the recognition algorithm iii) the deposition strategies adopted for the different features. Finally, the proposed CAM is tested on an actual component, highlighting its capability of dealing with the different features.

2. Proposed CAM software: overview

This paper proposes novel CAM software dedicated to thin-walled components manufactured through Wire Arc Additive Manufacturing (WAAM) technique. Examples of such parts are the lightweight aerospace components, such as wing ribs, stiffeners or landing gear components [5]. To achieve adequate structural stiffness, these components have several stiffeners often intersecting each other and originating "crossings". Such crossings can have various shapes which can be classified as specified in Table 1.

Table 1 Classification and schematic view of T-Crossing and Direct Crossing features

T Crossing		Direct Crossing	
3D View	Top View	3D View	Top View
			

The crossing features are critical, since they may require internal fillets (Fig. 2) to reduce stress concentrations, assure adequate fatigue strength and accessibility during milling operations, while porosity and flaws in these zones must be avoided for the same reasons.

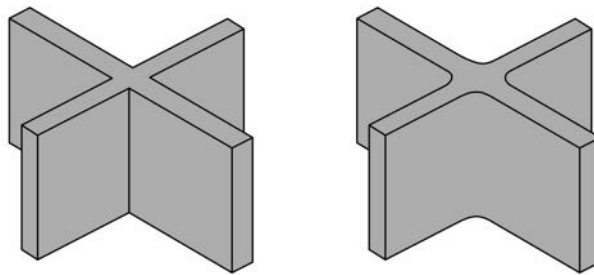


Fig. 2. Direct Crossing with no internal fillets (a) and with internal fillets (b).

Suitable crossings deposition strategies must hence be developed to prevent the occurrence of such defects. This paper proposes and compares several deposition strategies for crossings and tubulars; besides, it shows an implementation of these strategies in the proposed CAM software. This CAM software is completely automatic, since the input required to the user is just the selection of the CAD model. Since the software uses a single welding pass strategy (i.e. each layer is manufactured without bead overlapping), the input CAD model is composed just by the skeleton surfaces (i.e. midsurfaces) in an IGES 144 format. Therefore, it is not important to draw the exact final shape of the part including all the details such as the internal fillets of the crossings. In the presented CAM software, the wall thickness of the part is assumed as constant and its value is an input provided by the user, as explained in Section 3. Fig. 3 compares the full CAD model of a thin walled part and the skeleton surfaces required by proposed CAM software.

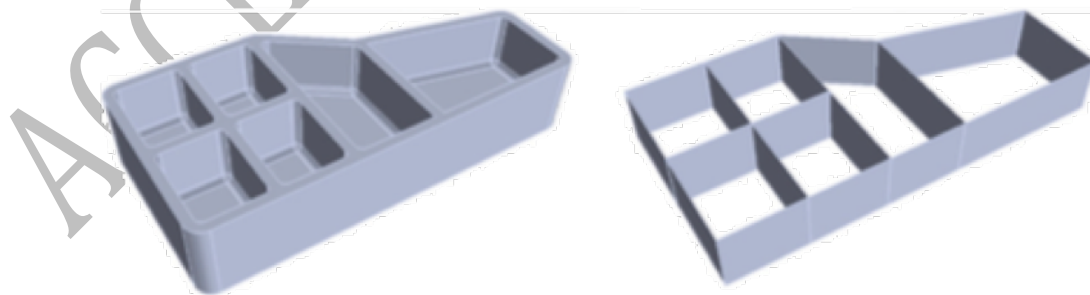


Fig. 3. The complete CAD model (left) and the midsurface-only CAD model (right) for a stiffened panel.

The proposed CAM software is composed by three modules, as shown by the schematic overview depicted in Fig. 4. Every module is described in detail in the following sections of this paper.

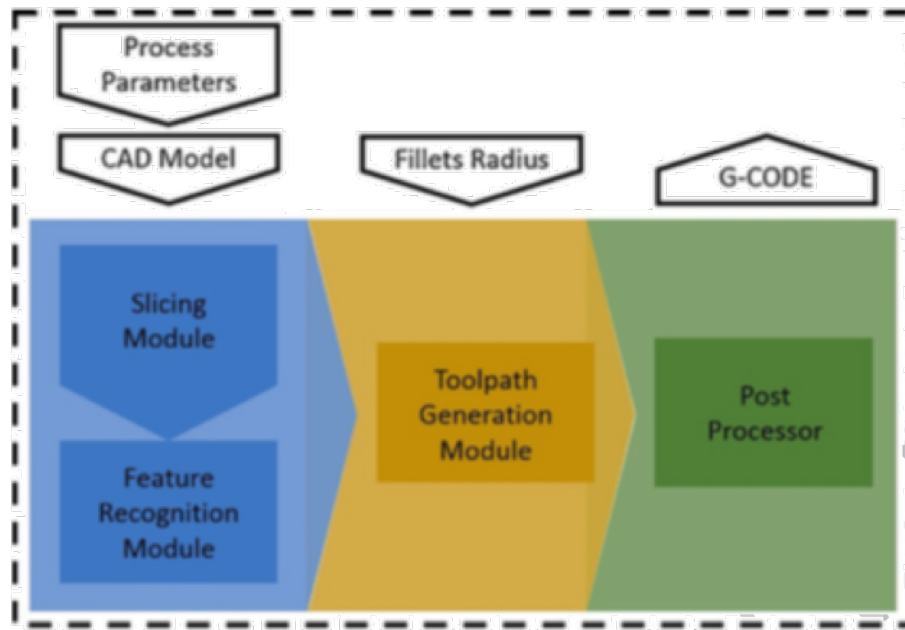


Fig. 4. The three modules composing proposed CAM software.

The first module is the Feature Recognition one, which takes the CAD model of the final part as input, recognizing and classifying all the different features of the component. The Toolpath Generation module then computes the deposition path by performing several operations, such as the generation of additional segments to deposit extra material in the corners of the crossings, ensuring a fair amount of stock allowance to machine the fillets. The radius of such fillets is an input that the user must give to the software (Fig. 4). At this stage, the software creates a Cutter Location File (CL File), which contains all the points of the toolpath in the coordinate system of the part. Finally, a post processor translates the CL file into a G Code with G01, G02 blocks and WAAM specific M instructions such as torch ignition and shut-off. G04 instructions are also used in the G Code to keep the axes of the WAAM NC machine in an idle state before starting the deposition while the arc is igniting.

The following sections of the paper detail each module of the CAM software together with the deposition strategies for crossings and tubular shapes. The algorithms depicted in this paper are described without referring to a specific programming language. Only the fundamental logic concepts and logical operations are presented. This approach enables the implementation and reproduction of the proposed CAM software with different programming tools and languages. For the testing and validation sake, the authors implemented the presented algorithms using a graphical programming language (or G Language) through the software Grasshopper [17] that is a plugin for the Rhinoceros CAD software [18] and uses several of the OpenNurbs [19] standard functions and logics.

3. Proposed CAM software: input description

The input format of the workpiece geometry is an IGES 144 file. This because it is a well-known interexchange format that can be read and produced by every CAD software suite. Besides, many literature works dealing with the analysis and processing of such format are available [20–22], together with algorithms under the Open Source License format. Moreover, since the proposed CAM software is tailored for thin walled components to be manufactured by stacking a single bead over the other, the midsurface, is the only required geometrical input. Therefore, an input CAD model for the proposed CAM software can be obtained by drawing the sketch of the midsurface base profile and extruding every curve of it along the building direction (i.e. usually the Z+direction). This returns a midsurface-only CAD model with the desired height (Fig. 5).

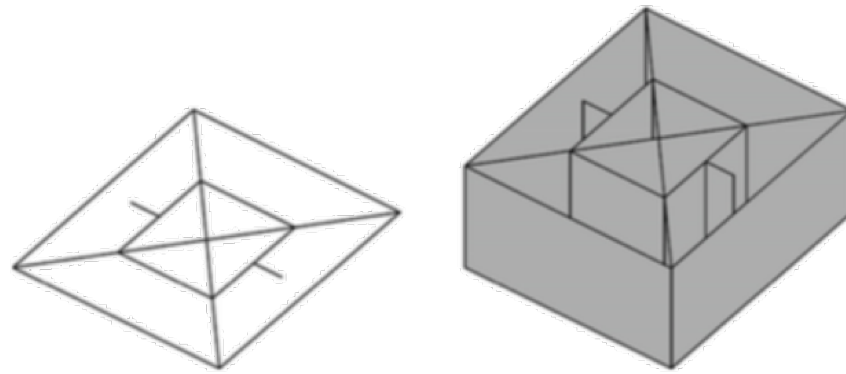


Fig. 5. Base curve (left) and midsurface-only CAD model obtained extruding the base curve (right).

It is important to notice, for a good comprehension of the following sections, that an IGES file as the one represented in Fig. 5 is composed by several surface entities, as shown in Fig. 6.

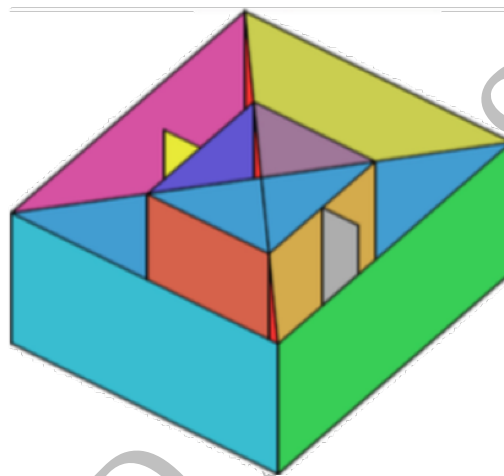


Fig. 6. The several surfaces composing the input CAD model.

It is important to point out this aspect since, due to this re- presentation of the surfaces, the single entities composing the final CAD geometry are processed individually in the software, for both feature recognition and toolpath generation.

Since the developed CAM software is totally automated, it can also be embodied in an application without a graphic area to view the input part or CAD tools to modify it. For this reason, the input geometry must be modeled assuming that the origin of the CAD software frame of reference will be the same of the deposition path. The work offset of the WAAM machine will be accordingly adjusted. In addition to the CAD model, the proposed CAM software requires further values to calculate the deposition toolpath:

- The Effective Wall Width (EWW).

As stated in several pieces of literature dealing with direct energy deposition techniques [23], two different values for the wall thickness can be identified: the Effective Wall Width (EWW) and the Total Wall Width (TWW). EWW is the maximum wall width after the finishing operations, achievable for a given set of welding parameters. TWW is the thickness of the wall just after the deposition, i.e. without any post processing. Fig. 7 clarifies the distinction between TWW and EWW.

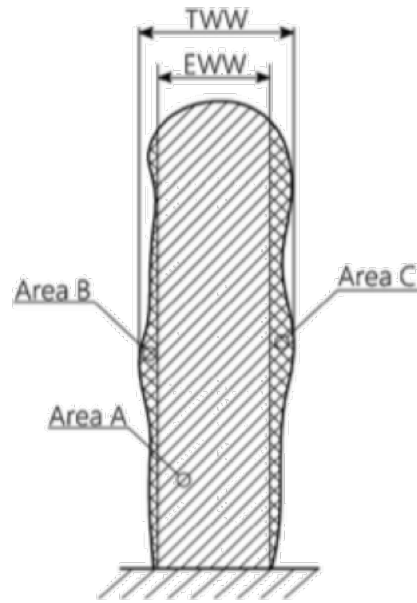


Fig. 7. Definitions of Effective Wall Width and Total Wall Width. Adapted from Jhavar et al. [23].

As earlier mentioned, such values depend on the set of deposition parameters and must be determined by experiments, according to the procedure reported in [24]. It can be noticed that the most desirable condition is to achieve an TWW as close as possible to the EWW to maximize the deposition efficiency DE that is defined as (Eq. (1)) [23]:

$$DE = \frac{AreaA}{AreaA + AreaB + AreaC} \quad \text{Equation 1}$$

- The value of the radius of the internal fillets of the crossings.

For the presented CAM software, it is assumed that all the internal fillets in the part have the same radius.

- The layer height (LH).

This input is used to perform the slicing operation, the starting point to obtain the toolpath from the input surfaces of the CAD model. The layer height depends on the weld bead height, which is related to the welding parameters. It can be determined through experiments in which several single beads and walls are built. The height of the single bead manufactured using a specific and repeatable set of welding parameters can be measured using a laser scanning device or a profil-ometer [24]. This procedure returns a first attempt value of LH. To achieve a more accurate estimation, a wall with a fixed number of layers can be manufactured using specific condition of interpass tem- perature. At the end of the manufacturing phase, an average LH value can be obtained by measuring the total wall height and dividing it by the number of deposited layers.

4. Proposed CAM software: the feature recognition module

This section details the Feature Recognition Module, which is the first part of the workflow of the proposed CAM software. This first module also includes the slicing algorithm to obtain the deposition toolpath form the input surfaces.

After the input geometry has been imported in the CAM environment, it is decomposed in its basic entities as depicted in Fig. 8. From a software point of view this means that every surface entity is extracted from the original input file and inserted in a list to be further processed in the next steps.

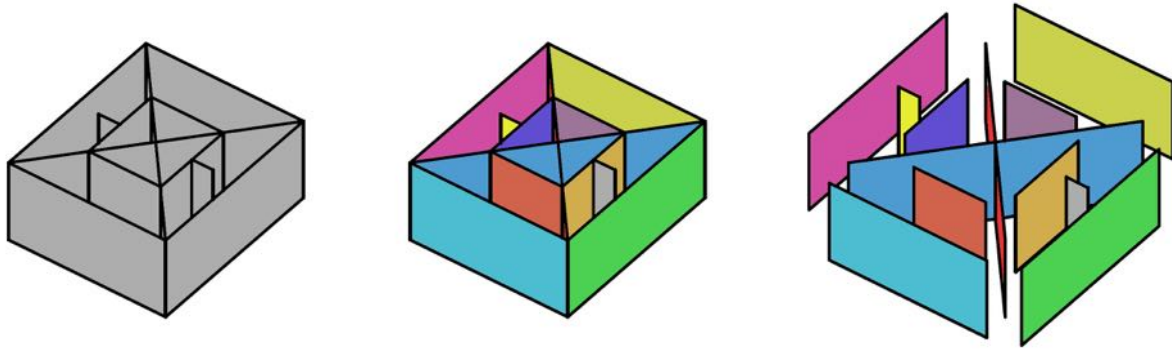


Fig. 8. Input midsurface-only CAD model (a) and the several surfaces composing it (b and c).

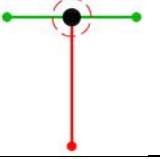
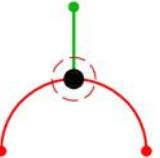
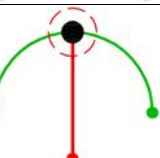
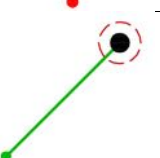
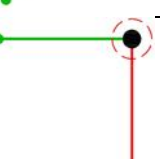
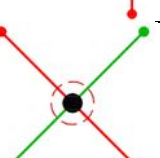
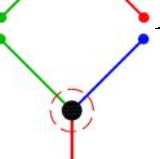
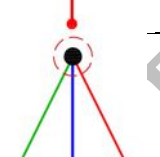
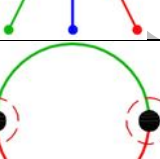
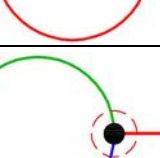
The subsequent step is the slicing procedure of every single item in the list. To perform such operation a surface-to-surface intersection operation must be carried out (an algorithm could be found for example in [25]). The intersection operation is performed between a plane with the normal oriented according to the building direction, i.e. the z-axis (0.0,0.0,1.0), and with a distance from the base plane equal to the layer height (determined by the user). The result of this step is a list of curves, each derived from a surface in the input IGES file. However, such list, called “First Layer Curve List”, represents the preliminary deposition toolpath for the first layer that will be further processed to obtain an optimized deposition strategy, especially in the zones of the crossings. All the operations of the Feature Recognition Module and Toolpath Calculation will be performed on such list. At the end, when the optimized toolpath has been generated for this list of curves, all the curves in the list itself are repeated in the Z direction a number of times equal to the height of the part divided by the layer height (the result is rounded to the closest higher integer). The further step is the feature recognition procedure itself. In this skeleton representation, the crossing features are identified as points (Fig. 9).



Fig. 9. Crossing features (cross marks) as identified in the CAD skeleton re-presentation.

This stage identifies and classifies the central point of each different feature in the model. The proposed algorithm is based on the specifically developed features taxonomy, depicted in Table 2.

Table 2 The Intersection Taxonomy developed by the authors on which the Feature Recognition procedure is based

Intersection Topology	“Valence for Intersection”	“Valence for End Points”	Conventional Name of Feature	Entities
	2	1	Straight T-Crossing	2
	2	1	Open T-Crossing	2
	2	1	Closed T-Crossing	2
	0	1	Free End Wall	1
	2	2	Corner Junction	2
	1	0	Direct Crossing	2
	6	3	Open T-Crossing	3
	6	3	Closed T-Crossing	3
	2	2	Continuous Junction	2
	6	3	Odd T-Crossing	3

First, every curve in the list is reparametrized to ensure that the curve parameter ranges from 0.0 to 1.0. At this stage, every curve of the list is intersected with all the other ones and the resultant points are stored in an array named “Intersection Strategy Array”. Then the end-points of every curve in the

list (i.e. the points corresponding to a curve parameter value of 0.0 and 1.0) are extracted and stored into another array named “End-Points Strategy Array”. All the steps described up to this point are schematically outlined in the flowchart of Fig. 10.

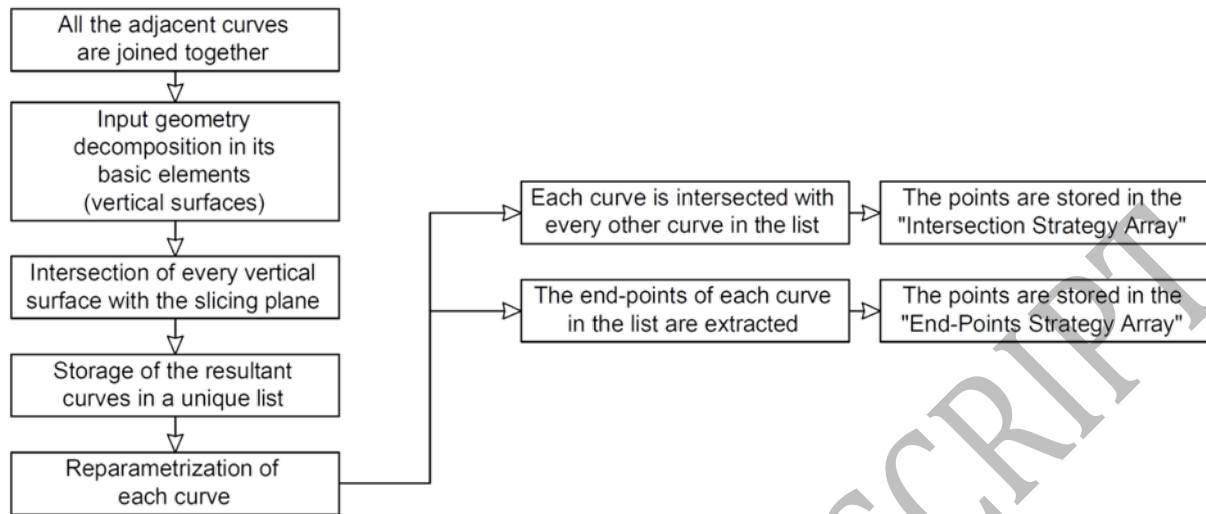


Fig. 10. Flowchart explaining the procedure to obtain the “End-Points Strategy Array” and “Intersection Strategy Array”.

When the procedure to extract the points terminates, the feature recognition algorithm starts. The base concept is that each crossing or feature is represented in the CAD space by a point. The result of the algorithm is hence an array of points in which every item represents the center of a different crossing/feature and will be one of the inputs for the further toolpath calculation procedure. The first step to locate the features is to cull the eventual duplicate points in the “Intersection Strategy Array” and in the “End-Points Strategy Array”. The duplicated/triplicated points are identified as the array elements having a distance between each other below a given threshold value. In this paper, the threshold value was set to $1.0e-6$ mm. At the same time the number of occurrence of every deleted duplicated point (i.e. the valence) is stored in another array, respectively one for the “Intersection Strategy Array” array called “Intersection Valence Array” and one for the “End-Points Strategy Array” array called “End-Points Valence Array”. This means that if there are three equal points in the array only the first one is kept and the number “Three” is stored in the valence array. Therefore, the elements of the strategy and valence arrays have a one-to-one relation: the former contains the point coordinates while the latter contains the number of occurrences in the original strategy array. The first of the further steps is the detection of the points with valence equal to two in the “End Points Array”. Indeed, referring to the taxonomy of Table 2, such points represent two adjacent walls or a simple sharp corner. Therefore, they are not a critical feature since they can be deposited using a continuous toolpath along the curves obtained directly from the slicing procedure. However, to enable the user to check the correct functionality of the software, the implementation presented in this paper highlights such points with a small red sphere in the graphic area, as shown in Fig. 11.

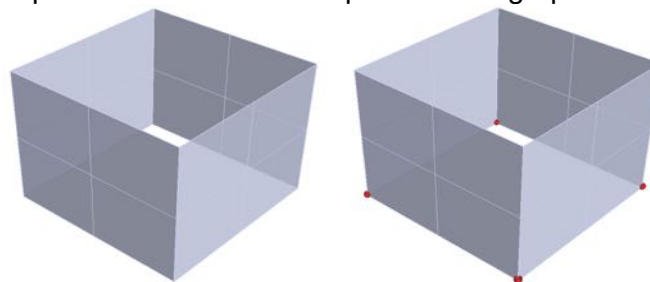


Fig. 11. The small red spheres highlight a continuous junction between two adjacent walls.

To identify the “Free End Wall” features, it is necessary to compare the two valence arrays, since the points representing such feature appear only once in the “End Points Strategy Array” but do not

appear at all in the “Intersection Strategy Array”. Once that such points are detected, they are stored in another array, i.e. the “End-Wall Array”.

Despite the “Free End Wall” might not seem a critical feature, it is important to locate them since, as highlighted by previous works [26], a deposition defect is present at both ends of such features, as shown in Fig. 12.

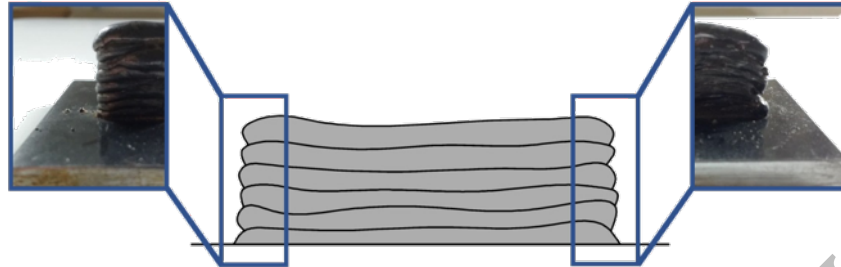


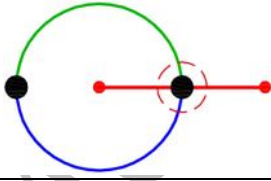
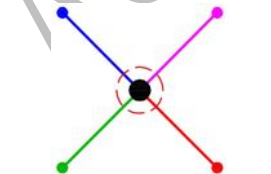
Fig. 12. Defects at the ends of a WAAM-manufactured wall.

To prevent this issue to affect the quality of the final workpiece, the toolpath must be extended with respect to the wall length, enabling to remove the defective regions during the milling phase.

At this stage, the already detected free end walls, the continuous junctions and all the points appearing only once in the “End Points Strategy Array” are erased from the array itself that will now contain only the points with a valence equal to three; therefore, referring to Table 2, these points represent “Open T-crossing”, “Closed T-Crossing” or “Odd T-Crossing” features. These points are stored in a dedicated array. The detection of the “Direct crossing” features is performed using a similar procedure to the one presented for the free end walls, i.e. comparing the “End Points Strategy Array” and the “Intersection Strategy Array”. That is because a direct crossing appears only once in the “Intersection Strategy Array” but is not present at all in the end points one. The procedure is slightly different for the straight T-Crossing. Indeed, it is necessary to consider the original “End Point Strategy Array” and eliminate from this the already detected points representing the continuous junctions, Open/Closed T-Crossings and Free End Walls. The points that remain in the array represent the straight T-Crossings.

Finally, there is a further case, which this paper refers to as “Odd Direct Crossings”, that must be considered to obtain a comprehensive feature recognition. The “Odd Direct Crossings” are presented in Table 3.

Table 3 Odd Direct-Crossing and Odd T-Crossing topology

Intersection Topology	Valence For “Intersection Between Curves”	Valence For “End Points”	Conventional Name of the Feature	Entities
	6	2	Odd Direct Crossing	3
	12	4	Four Direct Crossing	4

Such features would not be identified by the recognition strategy presented so far since the points representing them would be confused with the Continuous Junction. To detect these additional cases, it is necessary to create an array containing the points of the “Intersection Strategy Array” with valence six and an array with the points of the “End-Points Strategy Array” with valence two. Referring to Table 2 and to Table 3, it is possible to detect the points representing the “Odd Direct Crossings” and “Odd T-Crossings” by comparing the two resultant arrays. In Table 3 another case

is also reported; this is the “Four Direct Crossings” feature, composed of a respective number of entities (i.e. four vertical surfaces). To detect such feature, it is necessary to look for the points with valence 12 (Direct Crossing) in the “Intersection Strategy Array”.

The outcome of this recognition phase is a series of arrays for each type of feature. These arrays contain the central point of the features themselves. It is hence possible to use a color coding system to highlight the different features in the graphic area of the CAM software to advise the user whether every feature has been detected by the software. The result of this operation is exemplified in Fig. 13.

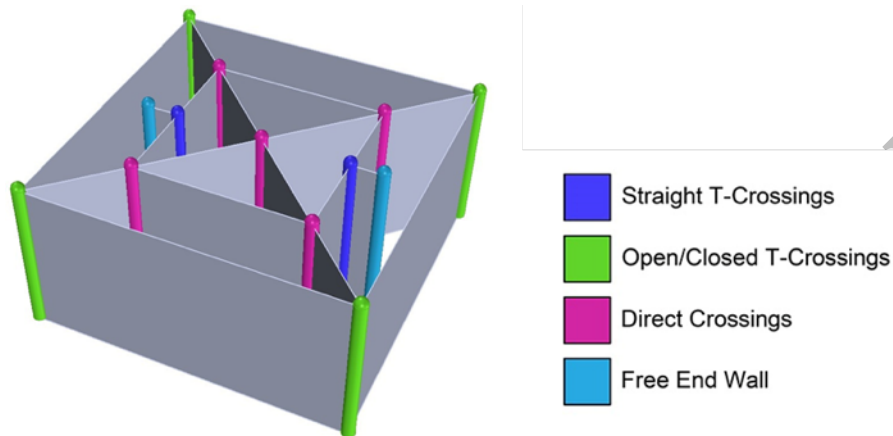


Fig. 13. Result of the feature recognition procedure: every different feature is highlighted using a different color.

To conclude the description of the feature recognition module a further step is needed to address situations such as the one presented in Fig. 14, where two separated parts are built on the same substrate.

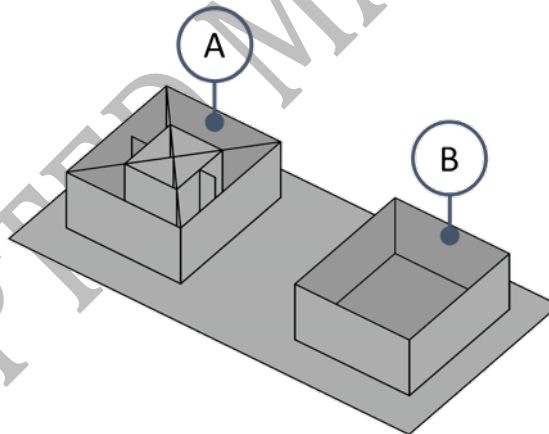
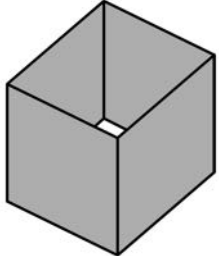
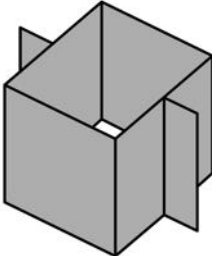
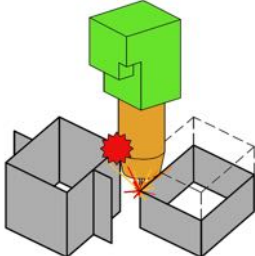
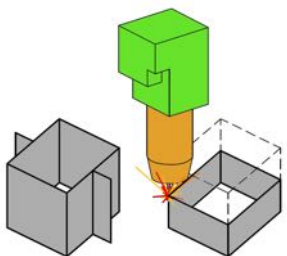


Fig. 14. A connected shape (A) and an isolated tubular (B).

Fig. 14 highlights that the part A has a complicated geometry which requires features recognition to achieve an optimal toolpath. On the opposite, the part B is a tubular shape without any crossings. It could be hence built by using a helical deposition strategy, i.e. generating a toolpath which is a helixlike curve having a base profile shaped as the tubular cross section and a pitch equal to the layer height. The next section clarifies that the helical strategy is the best suited to create tubular shapes. However, the helical approach can only be applied if no collisions between the different parts present on the substrate occur during the manufacturing procedure. Table 4 clarifies the concept of “isolated tubular”.

Table 4 Topology of isolated tubulars

Case 1	Case 2	Case 3	Case 4
--------	--------	--------	--------

			
<p>The tubular is alone on the building plate: the tubular is "isolated"</p>	<p>The tubular is connected to other geometries: the tubular is "NOT isolated"</p>	<p>During the building of the tubular a collision happens between the torch and an already existing part: the tubular is "NOT isolated"</p>	<p>During the building of the tubular no collisions happen between the torch and an already existing part: the tubular is "isolated"</p>

From a software perspective, to know whether an isolated tubular is present or not in the build plane (i.e. in the input geometry) the Feature Recognition Module of the CAM software acts as explained in the following lines and in Fig. 15. The starting point is the "First Layer Curve List" containing all the curves resulting from the slicing procedure. Every curve in the list undergoes a test to understand if it is closed; if a curve is closed, it is stored in an array called "Closed Curve Array" and deleted from the "First Layer Curve List", otherwise the software goes to the next curve. All the end-points of the curve remained in the "First Layer Curve List" are extracted and stored in a "End-Points Array".

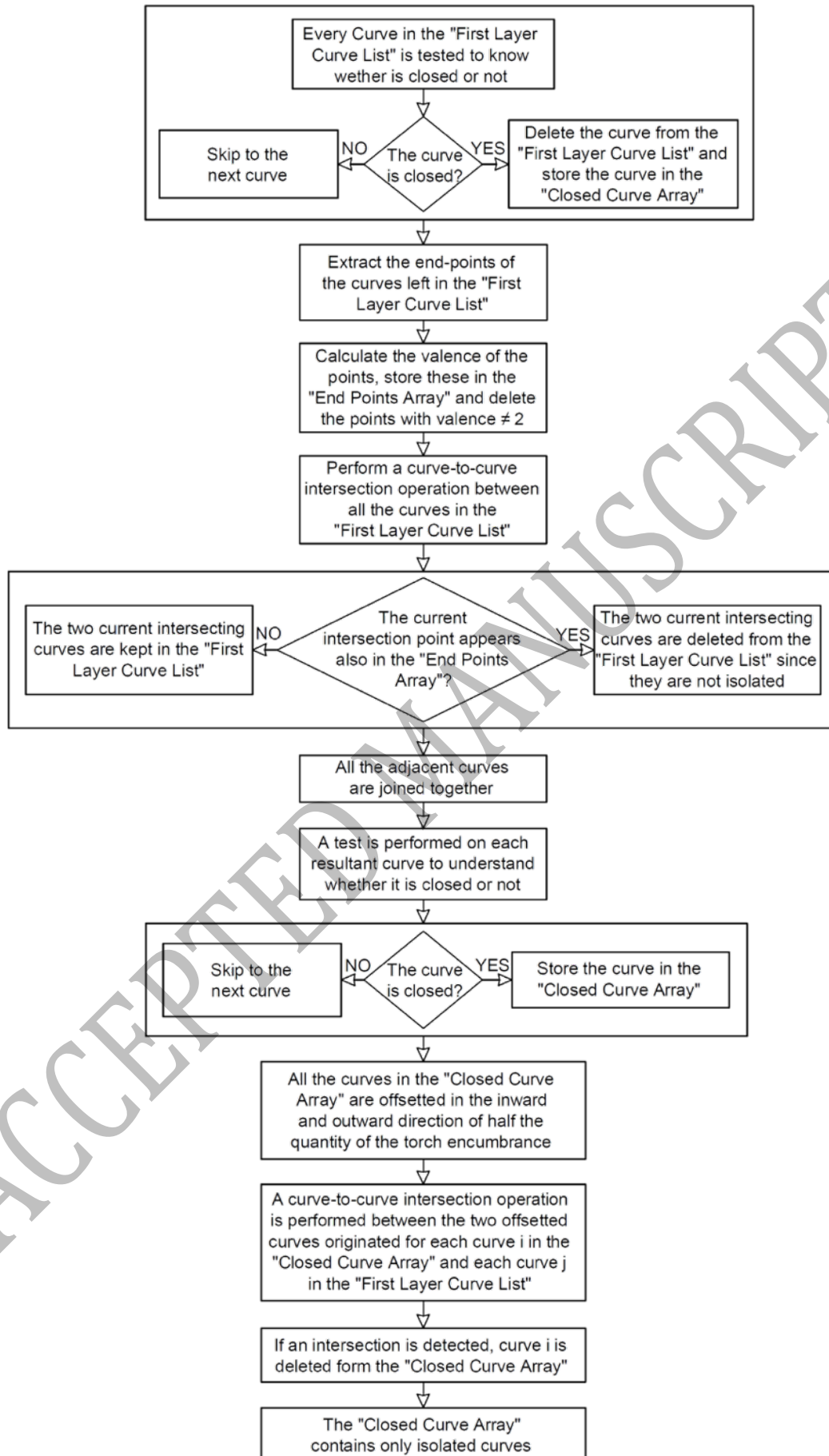


Fig. 15. Flowchart of the algorithm developed to detect isolated tubulars.

The valence of each point (i.e. how many times the same point appears in the array) is calculated and only the points with valence two (i.e. the points pinpointing the continuous junctions referring to Table 2) are kept in the array. Then, a curve-to-curve intersection operation is performed between each curve remained in the “First Layer Curve List” and all the other ones in the array: each time an intersection point is found it is compared with the points with valence two; if the detected intersection point appears also in the array of the points with valence two, the considered curve and the curve intersecting it are kept because there is a continuous junction between these, otherwise the curve are deleted from the array because it means that they are not isolated. At the end of the curve-to-curve intersection operation all the sets of adjacent curves in the “First Layer Curve List” are joined together. Another test is carried out to detect if there are other closed curves after the joining operation; if the test identifies further closed curves, these are stored in the “Closed Curve Array”. This way all the isolated closed curves (i.e. every base curve of a tubular-shape part) are detected. It is now important to understand if it is possible to use a helical strategy to build the part. Of course, if such strategy is used, the parts in the building plate are not created all together layer by layer: the tubular parts are built prior or subsequently to the other ones. This is only possible if no collision occurs between the torch and other parts in the building plate, because the separate deposition of the tubular part could lead to collisions between the torch and the already manufactured parts. To detect the presence of possible collisions, a further step is needed: the isolated curves stored in the “Closed Curve Array” are offsetted in the inward and outward direction of half the quantity of the maximum encumbrance of the torch; then, a reciprocal curve-to-curve intersection operation is performed between the offsetted curves originated from the generic i^{th} curve in the “Closed Curve Array” and all the other original curves that are present in the same “First Layer Curve List”. If any intersection is detected it means that the tubular related to the i^{th} curve cannot be built separately from the other parts using a helical strategy, but must be built layer-wise to avoid collisions. For this reason, it is deleted from the “Closed Curve Array” (Fig. 16).

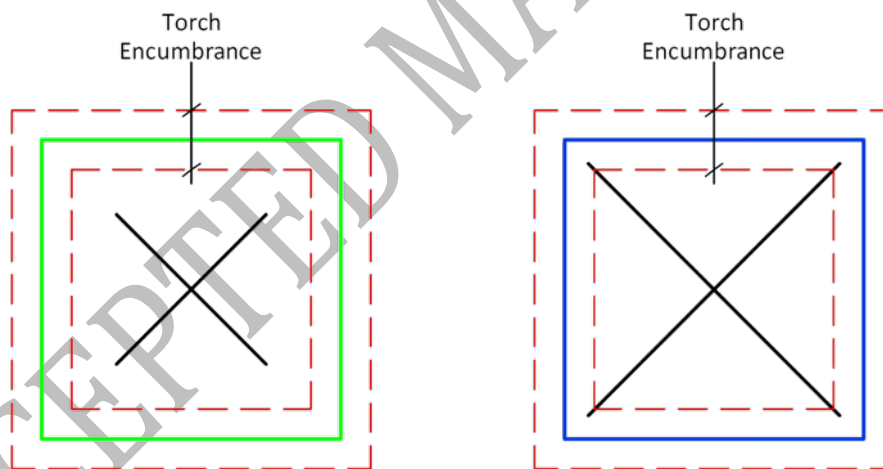


Fig. 16. Example of isolated tubular (left, green) and not isolated tubular (right, blue).

5. Proposed CAM software: deposition strategies

At this stage of the workflow, the software has generated: i) a preliminary deposition toolpath for the very first layer ii) a series of points subdivided according to the kind of feature they represent iii) a list of isolated curves representing the tubular features that can be built using a helical strategy. The software must now calculate the deposition toolpath specific for every kind of feature. The different deposition strategies implemented in the proposed CAM software are either taken from literature or developed by the authors. This section presents and details the deposition strategies available for every different feature together with some experimental results to highlight their effectiveness.

5.1. Deposition strategies for direct crossing

Several literature works presented deposition strategies for Direct crossings. For example, Uglia et al [27] and Mehnen et al [28] presented deposition strategies developed and tested specifically for the WAAM process, hence suitable for the implementation in the proposed CAM software. The most relevant deposition strategies for Direct Crossings developed in the aforementioned papers are presented in Fig. 17.

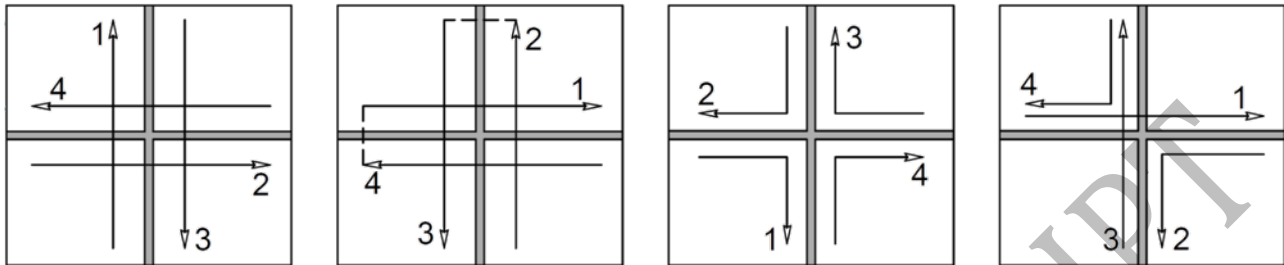


Fig. 17. Deposition strategies for Direct Crossings (a) and (c) from [28], (b) and (d) from [27].

Despite their effectiveness, all these strategies require each bead of the direct crossing to be deposited in a specific sense. The implementation of such strategies is hence straightforward only in the case of a single crossing. When dealing with complex workpieces which include many crossing features, using this approach would result in significant complications in the implementation phase. Moreover, the deposition sense constraint could lead to an impossible solution in joining multiple features. Since the goal of the CAM software is to deal with actual components, the authors developed a deposition strategy for Direct Crossings to overcome the issues of the literature ones. The proposed strategy is depicted in Fig. 18.

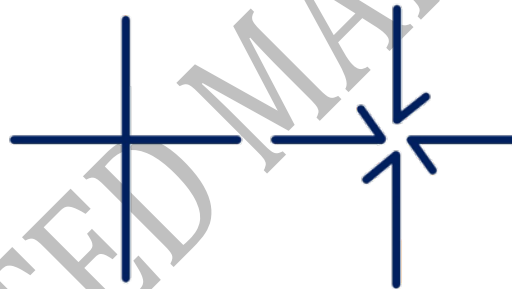


Fig. 18. The deposition strategy for Direct Crossings developed by the Authors. Original curves resulting from slicing operation (left) and toolpath for the direct crossing after processing (right).

From a software point of view the toolpath for this deposition strategy can be obtained with the four steps summarized in Fig. 19.

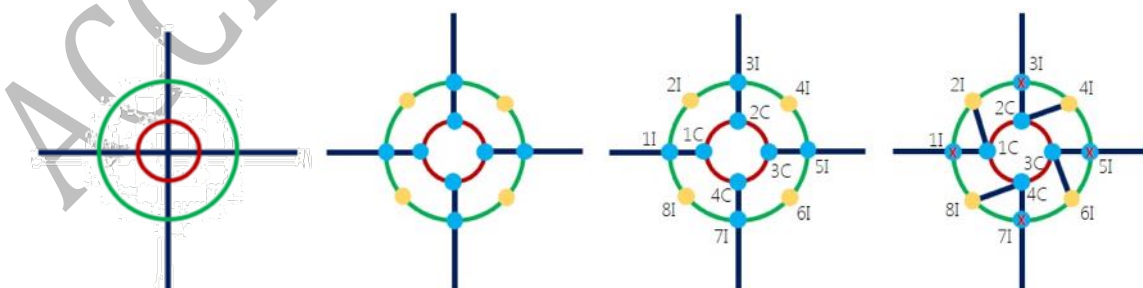


Fig. 19. The procedure to calculate the toolpath for the direct crossing implemented in the proposed CAM software.

Referring to Fig. 19, the deposition toolpath of a direct crossing is performed following the depicted steps:

- Two circumferences centered in the point resulting from the feature recognition step are generated; the inner one is called "Cutting Circle" and it is represented in red in Fig. 19(a); the outer one is called "Influence Circle" and is represented in green;
- The Cutting Circle is tested for intersection with all the curves of the "First Layer Curve List" and only the curves that intersect the circle itself are stored in a dedicated "Direct Crossing Curve Array". Moreover, the intersection points (cyan marks in Fig. 19(b)) are stored in a dedicated "Cutting Circle Intersections Array";
- The same operation is performed for the Influence Circle. The intersecting curves are the same that are already stored in the "Direct Crossing Curve Array"; on the opposite, the intersection points are different and are stored in an array called "Influence Circle Intersections Array";
- Four different arcs are obtained from the Influence Circle by splitting it at the points of the "Influence Circle Intersections Array", which are stored in the "Quarter Influence Circle Array". The midpoint of each of the four arcs is then calculated and stored in a new "Influence Circle Midpoints Array" (yellow marks in Fig. 19(b));
- The two curves composing the Direct Crossing are split at the points contained in the "Cutting Circle Intersections Array" and the portions of the curves included in the Cutting Circle are deleted. To detect such curves portions the midpoints of each of the curves obtained after the splitting operations are calculated and then tested for inclusion with respect to the Cutting Circle: if a midpoint is inside the Cutting Circle the correspondent curve portion is deleted (Fig. 19(b));
- The points in the "Cutting Circle Intersections Array" are ordered in a clockwise manner (Fig. 19(c));
- The points in the "Influence Circle Intersections Array" and in the "Influence Circle Midpoints Array" are all stored in a new array called "Influence Circle Toolpath Points Array" and ordered clockwise (Fig. 19(c)); it is important that the first point of the "Influence Circle Toolpath Points Array" and of the "Cutting Circle Intersections Array" lay on the same curve of the Direct Crossing as depicted in Fig. 19(c);
- A line is generated between the first point in the "Cutting Circle Intersections Array" and the second point of the "Influence Circle Toolpath Points Array" (Fig. 19(d)); such line is the first additional toolpath segment to deposit the extra material needed for the internal fillet of the Direct Crossing. This procedure is repeated until all the four additional segments have been generated.
- The additional segments are joined to the respective original segments of the Direct Crossing and the "First Layer Curve List" is updated.

It is important to highlight that the proposed strategy does not require any of the crossing beads to be deposited in a specific sense. Two important parameters to achieve a defect free deposition, i.e. with no voids embedded in core of the crossing are the radii of the influence and cutting circles. The theoretical values of such parameters are determined by geometrical consideration, as highlighted in Fig. 20.

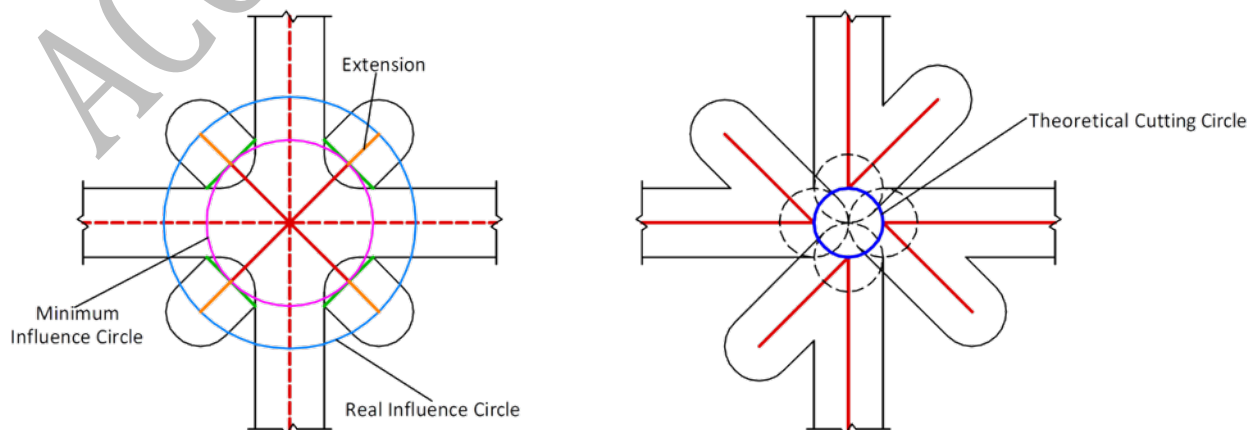


Fig. 20. Minimum value of the influence (a) and cutting (b) circle radii.

For what concerns the influence circle radius (Fig. 20(a)): given the skeleton surface of the crossings, the influence circle should be tangent to the four segments (one per each of the crossing branches) perpendicular to the skeleton line and having a length equal to the bead width (green lines in Fig. 20(a)). This results in the minimum influence circle marked by the pink line in Fig. 20(a). However, it is safer to extend the influence circle to locate the arc start/stop further from the core of the crossing, i.e. in a zone which is removed during the subsequent machining (cyan line in Fig. 20(a)). Moreover, this extension provides the machining allowance required to create the fillets during the finishing operation.

Regarding the cutting circle (Fig. 20(b)), its theoretical maximum size is defined as follows: given the four circles (one for each branch of the crossing) with diameter equal to the bead width, centered on the branches and tangent between each other (dashed black lines in Fig. 20(b)), the cutting circle (solid blue line in Fig. 20(b)) should intersect the centers of such four entities. However, to improve the quality of the crossing it is better to decrease such value since an excessively large cutting circle could lead to the formation of a void in the center of the crossing.

Considering the presented discussion, the theoretical values must be tested and tuned by performing experiments. In this paper, different combination of influence and cutting circle radii were tested on a sample direct crossing. All the parts manufactured during the present activity have been created using a commercial GMAW machine (AWELCO 250 PULSEMIG), whose torch has been fitted on a three-axis milling machine with a dedicated clamp. A single welding parameter set has been used for all the experiments and it is reported in Table 5.

Table 5 Process parameters used for the experiments

Current [A]	Voltage [V]	Deposition Speed [mm/min]	Wire Feed Speed [m/min]
80	18	300	4.6

The results for different values of such parameters are reported in Fig. 21.




Test 1	Test 2	Test 3
		
Influence Circle R: 7.0 mm Cutting Circle R: 2.5 mm	Influence Circle R: 8.0 mm Cutting Circle R: 2.0 mm	Influence Circle R: 10.0 mm Cutting Circle R: 2.0 mm

Fig. 21. Results of the sample direct crossings manufactured to show the importance of the values of the influence and cutting radii.

Test 1 was performed using the theoretical radii. It is highlighted that the crossing show both a void in the central zone and a lack of material in the fillet areas. Test 2 was performed by reducing the cutting circle radius and increasing the influence circle one. The parameters of Test 2 enabled to eliminate the central void but did not result in enough material for the fillets. Test 3 produced the best results, since a further increase of the influence circle radius provided enough machining allowance for the fillets, keeping the crossing without inner voids.

5.2. Deposition strategies for open, closed and straight T-crossings

The authors proposed and tested different approaches to deposit T-crossing features [14]. The outcome of this work was the selection of the optimal deposition strategies shown in Fig. 22.

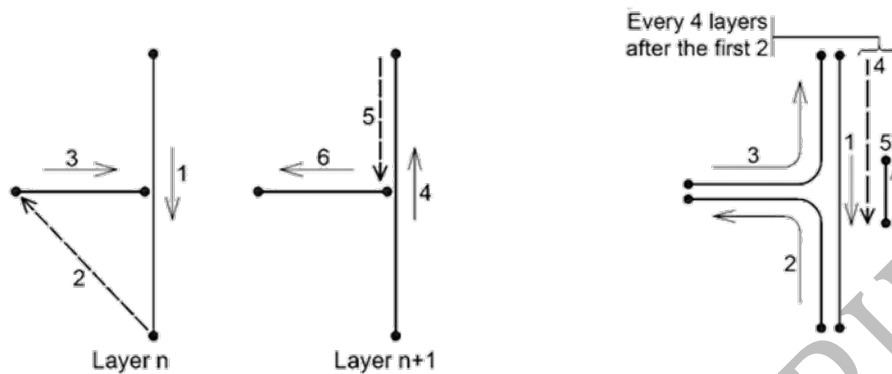


Fig. 22. Deposition strategies for T-Crossings with no internal fillets (left) and with internal fillets (right). From [14].

These strategies were developed by testing different approaches and selecting those which returned the best results in terms of the flatness of the back surface. The back surface of the T-crossing feature is depicted in Fig. 23.

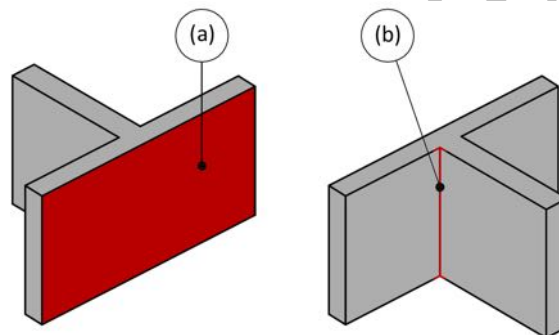


Fig. 23. Taxonomy for T-Crossing. (a) back surface, (b) internal corner.

This investigation was carried out manufacturing samples T-crossings, acquiring their back surfaces geometry via Coordinate Measurement Machine and then calculating the back surface flatness. However, as for the direct crossings, this strategy would require depositing the beads in a specific sense, complicating the software implementation or not resulting in a feasible path. To overcome this issue, the T-Crossings can be deposited using a strategy similar to the one developed for the Direct Crossings. Such strategy is depicted in Fig. 24 and it is the one implemented in the proposed CAM software.



Fig. 24. The new deposition strategy proposed by the Authors for T-Crossings. The original curves from the "First Layer Curve List" (left) and the toolpath optimized for the T Crossing (right).

Since the procedure is similar to the one proposed for the direct crossings, it is possible to refer to section 5.1 for details.

5.3. Deposition strategies for free end walls

As previously stated in the paper, it is important to extend the deposition toolpath in the zone of the free end walls to ensure an adequate machining allowance for the finishing operations. This is because arc ignition and extinguishing regions have an irregular geometry and can be affected by lack of fusion defects. Fig. 25 shows an example of toolpath extension. The amount of linear extension depends on the welding parameters and can be experimentally determined by measuring the defective zones to be removed depicted in Fig. 12.

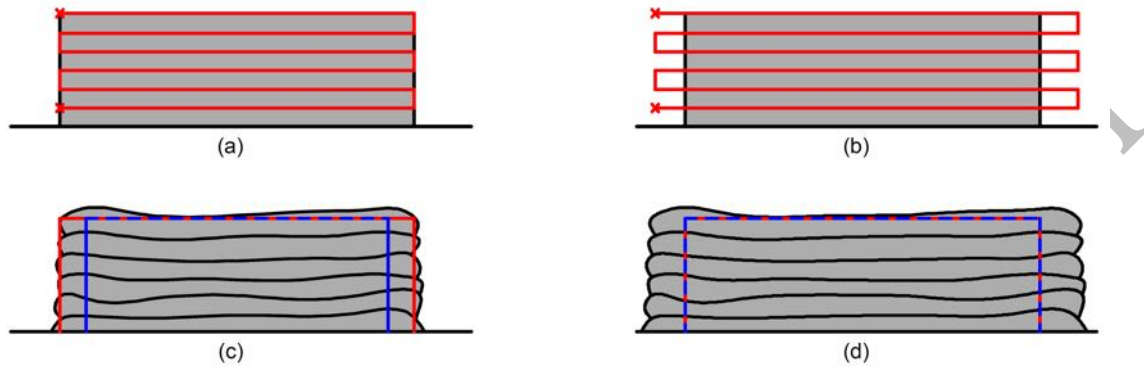


Fig. 25. The importance of toolpath extensions for free end walls for a straight wall (a) tool-path with no extensions, (b) tool-path with extensions, (c) expected result (red) and effective result (red) in the no-extensions case, (d) expected result (red) and effective result (blue) in the case of a toolpath with extensions.

5.4. Deposition strategies for tubulars

The goal of implementing tubular deposition strategies is to prevent the arc ignition and extinguishing defects to affect the flatness of the top surface [26]. Indeed, the region of the layer close to the arc ignition exhibits an excessive thickness with respect to its central part, due to the heat sink effect of the base metal. On the opposite, in the arc extinguishing region, the layer exhibits a reduced thickness due to the arc pressure on the molten pool. Therefore, if a tubular shape is deposited by igniting and extinguishing the arc in the same point for each layer, a deposition defect will be present on the surface. To overcome these issues, the authors propose three strategies: two of them use a layer by layer approach, while the other one uses a helical deposition strategy Fig. 26: depicts these strategies by showing an example of a tubular with circular cross section (i.e. a cylinder).

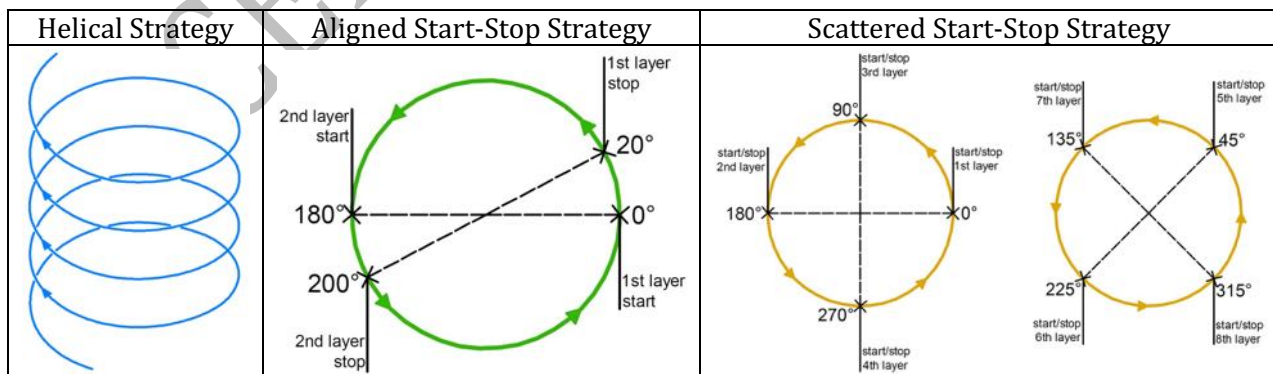


Fig. 26. Deposition strategies for tubulars.

The helical strategy is basically a continuous deposition pattern, which avoids inter-layer arc ignition and extinguishing. The aligned start-stop strategy aims at compensating for the defect by introducing a selfoverlap of the layer in the stop region. This way, the overlapping compensates for the lack of

material in the extinguishing region. The overlapping angle depends on the radius of the cylinder and can be calculated referring to Fig. 27 and Eq. (2).

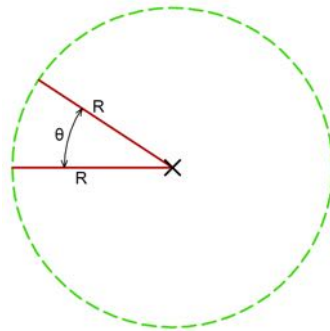


Fig. 27. Overlapping angle concept for Aligned Deposition Strategy.

The parameter O_l represents the amount of overlap between the two layers. This value can safely be set equal to the length of the arc crater for the given set of welding parameters. Finally, the scattered start-stop strategy distributes the start and stop points along the cross section.

This aims at preventing an excessive accumulation of defects in a specific point.

The effectiveness of such strategies was compared by manufacturing cylindrical tubulars using the three different approaches. The results were compared in terms of flatness of the top surface of the cylinder, i.e. by measuring its distance from the substrate in different points and analyzing the punctual deviation from the average height value. Fig. 28 presents the results of such comparison, while Fig. 29. shows the top surfaces of the tubulars manufactured using the three presented deposition strategies.

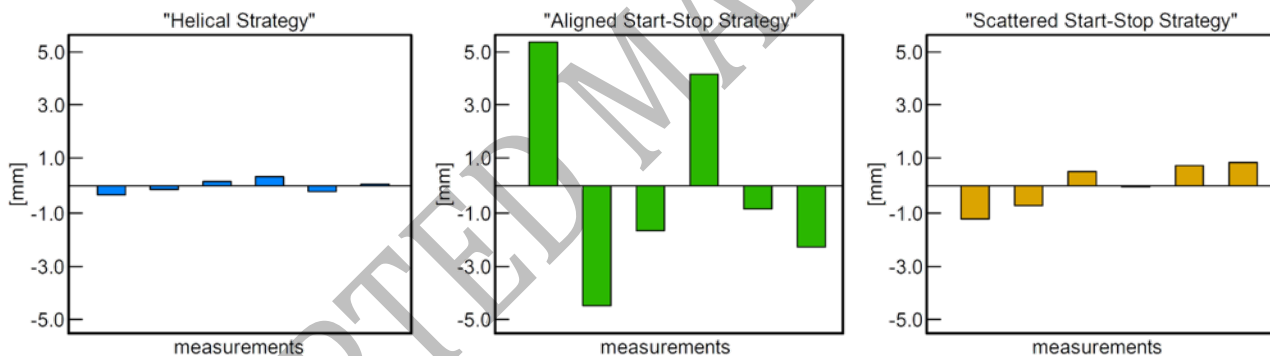


Fig. 28. Difference between the mean height and six measurements taken along the cylinder circumference to compare the different deposition strategies.

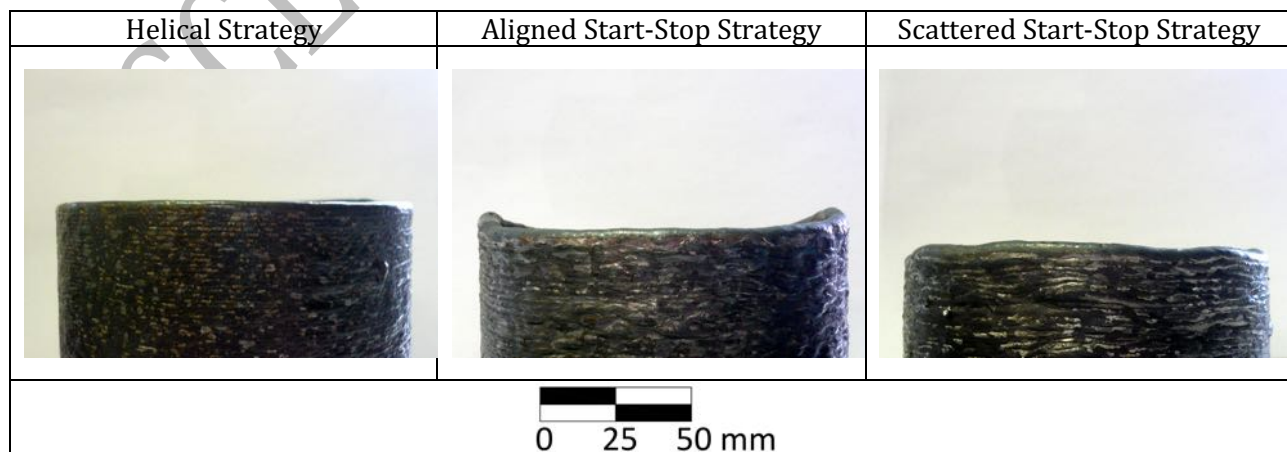


Fig. 29. Top surfaces for three different tubulars manufactured with the three presented deposition strategies.

It is highlighted that the “Helical Strategy” and the “Scattered Start-Stop Strategy” achieve the best results. Hence, these are the strategies implemented in the CAM software to deposit the tubular shapes. Using the layer by layer strategy (i.e. the “Scattered Start-Stop Strategy”) is mandatory in case of potential collisions when building multiple parts on the same substrate. However, it should be avoided as much as possible since, as shown by Fig. 28, it results in a reduced flatness with respect to the helical pattern. Moreover, it leads to many arc ignition phases, an always critical operation in WAAM since the arc could sometimes not ignite, compromising the manufacturing of the part. However, a layer-by layer approach could be necessary to introduce interlayer idle times to prevent the heat accumulation phenomenon. In this case a helical strategy would not offer any advantage respect to a layer by layer approach. It must be pointed out that, at the current state, the proposed CAM software is not aware of the process thermal issues. The eventual introduction of idle times is up to the user, which can use literature approaches to estimate their value [29].

6. Proposed CAM software: implementation and testing

The authors implemented all the algorithms presented in this paper in a software application using the graphic programming language (also known as “G Language”) of Grasshopper that is a plugin of the Rhinoceros CAD software. Grasshopper offers a software development environment, while Rhinoceros is used to manage the input CAD file (that can be produced using any other CAD software and exported in IGES 144 format) and display the toolpath and the outcome of the feature recognition module. Since the implemented solution uses Rhinoceros and Grasshopper, it is strongly dependent on the OpenNurbs standard. The workflow to produce the deposition toolpath for a part is described in the scheme of Fig. 30.

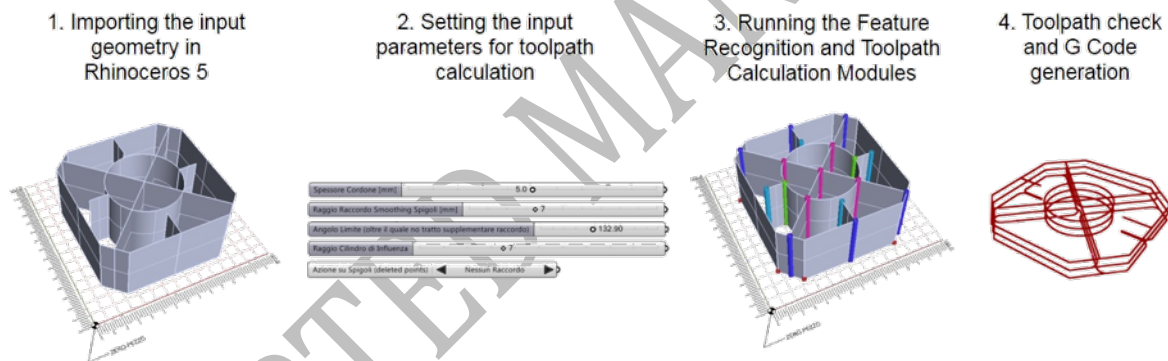


Fig. 30. The workflow of the proposed CAM software.

The developed CAM software and the feature recognition module were tested on several parts including the features reported in Table 2. Fig. 31 presents some examples of results produced by the feature recognition module.

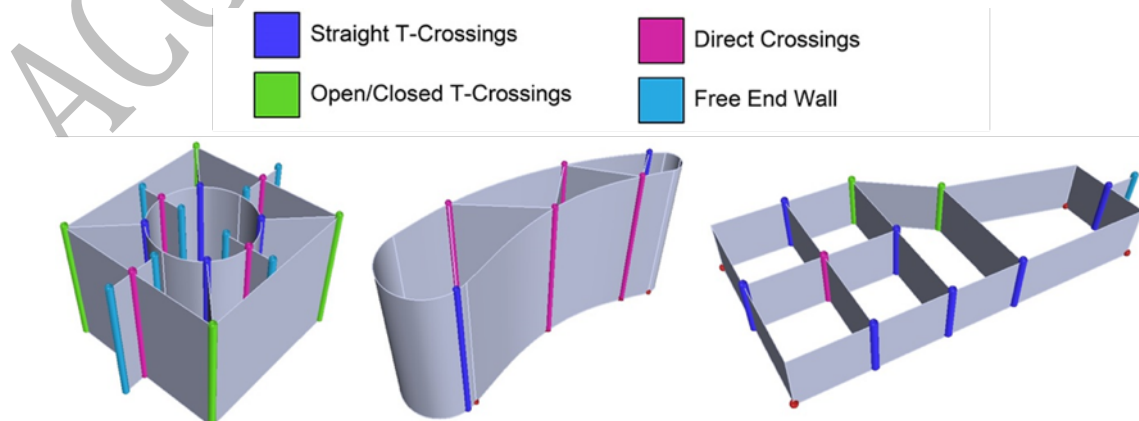


Fig. 31. The result of the proposed feature recognition routine on different geometries.

The result of these tests highlights that the proposed feature recognition algorithm can recognize all the depicted features. Besides testing the feature recognition algorithm, the proposed CAM software was used to generate the toolpath for the sample component, presented in Fig. 32.

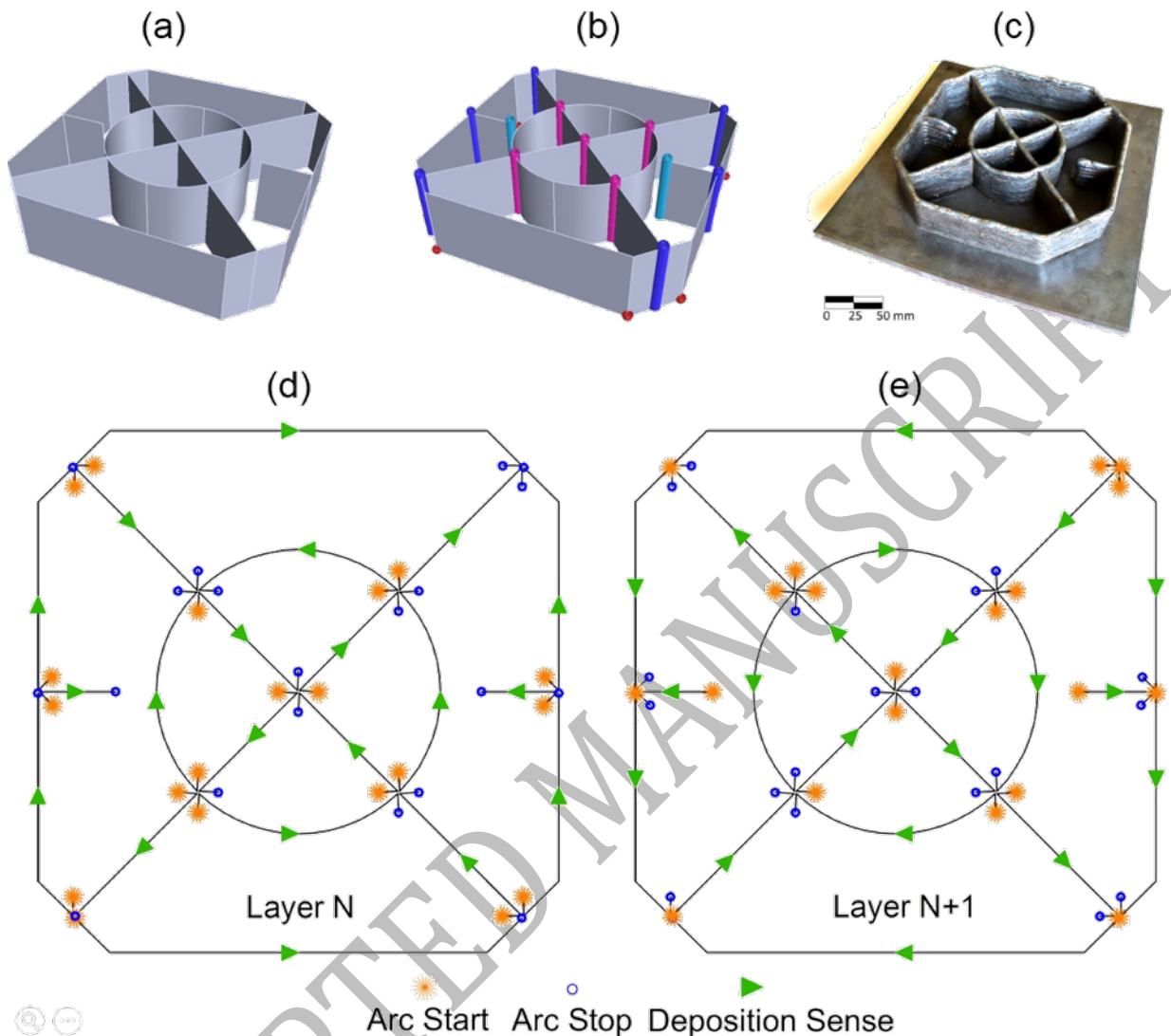


Fig. 32. The midsurface-only input CAD file (a); the result of the feature recognition module (b); the manufactured part before the finish machining operations (c); the arc ignitions and stops on a generic N layer (d); the arc ignitions and stops on a generic N + 1 layer (e).

Fig. 32(a) shows the midsurface-only CAD model, while Fig. 32(b) shows the outcome of the feature recognition. It is highlighted that the test case part has free end walls, direct crossings and tcrossings, i.e. all the features requiring dedicated deposition path. Fig. 32(d) and (e) show the calculated toolpath; arc start and stop points are highlighted. The test case was actually manufactured using the WAAM machine presented in [30]. Fig. 32(c) shows the manufactured test case in the as-welded state, i.e. prior to the finishing operation. It is highlighted that the toolpath generated by the proposed CAM software resulted in the desired shape.

7. Conclusions

This paper presents a novel approach to generate WAAM deposition path. Unlike previous works which propose to use multiple beads to create the cross section of the workpiece, this paper proposes a single pass approach, in which the cross sections of the part are deposited using a single welding bead. The advantage of this approach is twofold: it reduces the manufacturing time of thin-

walled parts and avoids dealing with bead overlapping, which requires a deep knowledge and control of the welding process. Hence, this paper proposes new three axis CAM software dedicated to the WAAM process. Since the single pass strategy could result in significant deposition errors when depositing crossing features, a feature recognition module is developed and implemented in the CAM software. The authors propose a taxonomy of the most critical features: free end walls, T-crossings, direct crossings and isolated tubulars. A feature recognition algorithm capable of detecting these features is developed. This way, the CAM software is aware of the workpiece criticalities and has the capability to modify the deposition path, triggering dedicated strategies for each one of the presented features. The deposition strategies are both derived from previous works and developed by the authors. The effectiveness of the proposed strategies is demonstrated by presenting the results of experiments. The overall effectiveness of the proposed CAM software is tested by running the feature recognition algorithm on sample geometries and manufacturing an actual test case component. It is highlighted that the proposed CAM software generates a suitable deposition path.

In conclusion this paper proposes an alternative technique to generate WAAM deposition path. It is worth to point out that the proposed single pass approach could not be the best suited for every situation. Indeed, depending on the specific welding process, achieving a high wall width with a single pass could result in an excessive heat input, detrimental for material properties and structural integrity of the workpiece. The aim of the paper is hence to propose an alternative approach, rather substituting the existing ones.

REFERENCES

- [1] C. V Haden, G. Zeng, F.M. Carter, C. Ruhl, B.A. Krick, D.G. Harlow, Wire and arc additive manufactured steel: Tensile and wear properties, *Addit. Manuf.* 16 (2017) 115–123. doi:10.1016/j.addma.2017.05.010.
- [2] M. Liberini, A. Astarita, G. Campatelli, A. Scippa, F. Montevecchi, G. Venturini, M. Durante, L. Boccarusso, F.M.C. Minutolo, A. Squillace, Selection of Optimal Process Parameters for Wire Arc Additive Manufacturing, *Procedia CIRP.* 62 (2017) 470–474. doi:10.1016/j.procir.2016.06.124.
- [3] K.F. Ayarkwa, S.W. Williams, J. Ding, Assessing the effect of TIG alternating current time cycle on aluminium wire + arc additive manufacture, *Addit. Manuf.* 18 (2017) 186–193. doi:10.1016/j.addma.2017.10.005.
- [4] F. Montevecchi, G. Venturini, N. Grossi, A. Scippa, G. Campatelli, Finite Element mesh coarsening for effective distortion prediction in Wire Arc Additive Manufacturing, *Addit. Manuf.* 18 (2017) 145–155. doi:10.1016/j.addma.2017.10.010.
- [5] A. Addison, J. Ding, F. Martina, H. Lockett, S. Williams, Manufacture of Complex Titanium Parts using Wire + Arc Additive Manufacture, in: *Titan. Eur.* 2015, 2015.
- [6] D. Ding, Z. Pan, D. Cuiuri, H. Li, A tool-path generation strategy for wire and arc additive manufacturing, *Int. J. Adv. Manuf. Technol.* 73 (2014) 173–183. doi:10.1007/s00170-014-5808-5.
- [7] D. Ding, Z. Pan, D. Cuiuri, H. Li, N. Larkin, Adaptive path planning for wire-feed additive manufacturing using medial axis transformation, *J. Clean. Prod.* 133 (2016) 942–952. doi:10.1016/j.jclepro.2016.06.036.
- [8] J. Kao, F.B. Prinz, Optimal Motion Planning for Deposition in Layered Manufacturing, 1998 ASME Des. Eng. Tech. Conf. (1998) 10.
- [9] Y. Nilsiam, P. Sanders, J.M. Pearce, Slicer and process improvements for open-source GMAW-based metal 3-D printing, *Addit. Manuf.* 18 (2017) 110–120. doi:10.1016/j.addma.2017.10.007.
- [10] Y. Cao, S. Zhu, X. Liang, W. Wang, Overlapping model of beads and curve fitting of bead section for rapid manufacturing by robotic MAG welding process, (2011). doi:10.1016/j.rcim.2010.11.002.
- [11] J. Xiong, G. Zhang, H. Gao, L. Wu, Modeling of bead section profile and overlapping beads with experimental validation for robotic GMAW-based rapid manufacturing, *Robot. Comput. Integr. Manuf.* 29 (2012) 417–423. doi:10.1016/j.rcim.2012.09.011.

- [12] D. Ding, Z. Pan, D. Cuiuri, H. Li, A multi-bead overlapping model for robotic wire and arc additive manufacturing (WAAM), *Robot. Comput. Integr. Manuf.* 31 (2015) 101–110. doi:10.1016/j.rcim.2014.08.008.
- [13] J. Mehnen, J. Ding, H. Lockett, P. Kazanas, Design study for wire and arc additive manufacture, *Int. J. Prod. Dev.* 19 (2014) 2. doi:10.1504/IJPD.2014.060028.
- [14] G. Venturini, F. Montecchi, A. Scippa, G. Campatelli, Optimization of WAAM Deposition Patterns for T-crossing Features, in: *Procedia CIRP*, 2016: pp. 95–100. doi:10.1016/j.procir.2016.08.043.
- [15] H.L. Lockett, M.D. Guenov, Graph-based feature recognition for injection moulding based on a mid-surface approach, *CAD Comput. Aided Des.* 37 (2005) 251–262. doi:10.1016/j.cad.2004.06.010.
- [16] S. Subrahmanyam, M. Wozny, An overview of automatic feature recognition techniques for computer-aided process planning, *Comput. Ind.* 26 (1995) 1–21. doi:10.1016/0166-3615(95)80003-4.
- [17] S. Davidson, Grasshopper (Algorithmic modeling for Rhino), (2015). <http://www.grasshopper3d.com/> (accessed March 14, 2018).
- [18] Robert McNeel & Associates, Rhinoceros 5 for Windows User's Guide, 2016. http://docs.mcneel.com/rhino/6/usersguide/en-us/windows_pdf_user_s_guide.pdf (accessed March 14, 2018).
- [19] SDK openNURBS, (n.d.). <https://www.rhino3d.com/opennurbs> (accessed March 14, 2018).
- [20] E.A. Nasr, A.K. Kamrani, Computer-Based Design and Manufacturing Computer-Based Design and Manufacturing, 2007. <http://www.springerlink.com/content/n2574k72123p2g24/abstract/>.
- [21] E. Reid, G. Morea, D. a. Harrod, A. Peltzman, C. Parks, W.B. Gruttke, The Initial Graphics Exchange Specification (IGES) Version 5.x, (2006) 1--754.
- [22] A.K. Kamrani, E.A. Nasr, Rapid Prototyping: Theory and Practice, 2006. doi:10.1039/c1c20514e.
- [23] S. Jhavar, N.K. Jain, C.P. Paul, Development of micro-plasma transferred arc (μ -PTA) wire deposition process for additive layer manufacturing applications, *J. Mater. Process. Tech.* 214 (2014) 1102–1110. doi:10.1016/j.jmatprotec.2013.12.016.
- [24] A. Adebayo, Characterization of integrated WAAM and machining process, Cranfield University, 2013.
- [25] X. Li, H. Jiang, S. Chen, X. Wang, An efficient surface – surface intersection algorithm based on geometry characteristics, *Comput. Graph.* 28 (2004) 527–537. doi:10.1016/j.cag.2004.04.008.
- [26] J. Xiong, Z. Yin, W. Zhang, Forming appearance control of arc striking and extinguishing area in multi-layer single-pass GMAW-based additive manufacturing, *Int. J. Adv. Manuf. Technol.* 87 (2016) 579–586. doi:10.1007/s00170-016-8543-2.
- [27] A.A. Uгла, O. Yilmaz, Deposition-Path Generation of SS308 Components Manufactured by TIG Welding-Based Shaped Metal Deposition Process, *Arab. J. Sci. Eng.* (2017). doi:10.1007/s13369-017-2582-3.
- [28] J. Mehnen, J. Ding, Design study for wire and arc additive manufacture, X (n.d.).
- [29] F. Montecchi, G. Venturini, N. Grossi, A. Scippa, G. Campatelli, Idle times selection for wire-arc-additive-manufacturing: a finite element based technique, *Addit. Manuf.* (2018).
- [30] F. Montecchi, G. Venturini, A. Scippa, G. Campatelli, Finite Element Modelling of Wire-arc-additive-manufacturing Process, in: *Procedia CIRP*, 2016: pp. 109–114. doi:10.1016/j.procir.2016.08.024.