Efficient toolpath planning for collaborative material extrusion machines

Efficient toolpath planning for collaborative material extrusion machines

Abstract

Purpose: Timing constraints affect the manufacturing of traditional large-scale components through the material extrusion technique. Thus, researchers are exploring using many independent and collaborative heads that may work on the same part simultaneously while still producing an appealing final product. This research proposes a simple and repeatable approach for toolpath planning for gantry-based n independent extrusion heads with effective collision avoidance management.

Design/methodology/approach: The research presents an original toolpath planner based on existing slicing software and the traditional structure of G-code files. While the computationally demanding component subdivision task is assigned to CAD and slicing software to build a standard G-code, the proposed algorithm scans the conventional toolpath data file, quickly isolates the instructions of a single extruder and inserts brief pauses between the instructions if the non-priority extruder conflicts with the priority one.

Findings: The methodology is validated on two real-life industrial large-scale components using architectures with two and four extruders. The case studies demonstrate the method's effectiveness, reducing printing time considerably without affecting the part quality. A static priority strategy is implemented, where one extruder gets priority over the other using a cascade process. Results demonstrate that different priority strategies reflect on the printing efficiency by a factor equal to the number of extrusion heads.

Originality/value: This research produces an original methodology to efficiently plan the extrusion heads' trajectories for a collaborative material extrusion architecture.

Keywords: Additive manufacturing; Collaborative manufacturing; Multiple heads; MEX; FDM, toolpath planning.

Paper type Research paper

Introduction

Design flexibility, a quicker design-to-production cycle, and reduced internal logistics are well-known benefits that contribute to the greater use of Additive Manufacturing (AM) as a primary manufacturing technique (Abdulhameed et al., 2019). Due to well-established advantages, the scientific and industrial interest (Murr, 2016) in AM have soared in recent decades. For example, the primary aerospace industries use AM to manufacture wind tunnel models, flight test parts, UAVs, engine parts, air ducts, wall panels and structural metal components, which are only a few examples of the possibilities (Ferro et al., 2016). Although used mainly for prototyping, in the automotive industry AM is employed in specialised applications such as motorsport and performance racing to manufacturing components modelled using generative design and topology optimisation tools, as well as to produce spare parts (Chinthavali, 2016).

However, major unresolved issues still keep AM from being the primary production technique in the industrial context. A non-inclusive list includes anisotropic ultimate product material qualities, CAD software limitations compared to AM's excellent design freedom, a restricted material portfolio, a lengthy

certification procedure (when there is one), expensive raw material prices, and high manufacturing costs for large batches (Guessasma et al., 2015). Compared to traditional manufacturing methods, repeatability and reproducibility are critical concerns in AM, mainly due to primary errors induced by a non-optimal selection of printing settings, which may influence the singularity rate (Ferretti et al., 2021). Furthermore, owing to the restricted manufacturing volumes available in off-the-shelf machines, the slow manufacturing process, and low production rates, the application of AM to large-scale component manufacturing nowadays is critical (Bacciaglia et al., 2022). Conventionally, large-scale objects could be defined as items with at least one of their three dimensions greater than one meter.

According to the standardised language of the ISO/ASTM 52900:2021 (*DIN EN ISO/ASTM 52900*, 2021), Material Extrusion (MEX) technology, also known as Fused Deposition Modelling (FDM), is commonly used to build prototypes because of the low costs of thermoplastic raw materials and tools. MEX can be defined as a material extrusion technique based on the deposition of polymer materials extruded through a nozzle in a layered pattern to reproduce the object's cross-section. To avoid the staircase effect (Espalin et al., 2014) and not affect the quality of the external skin appearance, each layer's thickness should be lowered, lengthening the production process. However, especially for large-scale components, the increase in production time may not be sustainable. Moreover, energy is lost for prolonged operations when heated bases or chambers are used. Though there are demonstrations of the employment of AM process to build large-scale components, such as the entire Unmanned Aerial Vehicles (UAVs) structure produced with FDM technology in aerospace applications, although the wing is split into many components due to the smaller maximum printing size (Pecho et al., 2019). Only minor components are manufactured in a single piece in the automotive sector, where the reference size for printing an entire car in AM is 4.5m. Thus, nowadays, a conflict of objectives, also called "trilemma" (M. Leite et al., 2018), affects AM technology: namely, a designer can have just two out of three specs between 1) small parts with high quality and low build time, 2) significant components with a prolonged process and high quality and 3) large-scale part with fast process and low quality (Marco Leite et al., 2018).

To address this open challenge, scientists are designing new collaborative frameworks capable of producing large-scale, high-quality components in a reasonable amount of time by combining some separate extrusion heads that collaborate on the manufacturing of the same component (Fontaine, 2016; Frutuoso, 2017; Marco Leite et al., 2018; Robotics; Wachsmuth, 2008). (Zhang et al., 2018) describes a 3D printing framework that employs multiple mobile robots that depose concrete material for large, single-piece structures. Zhang et al. use sophisticated and expensive sensors to avoid collisions between the swarm of AM machines. Moreover, low layer resolution typical of civil engineering applications, in the order of 10mm, affects the manufactured structures, thus limiting the application in aerospace and automotive components.

In contrast, (McPherson and Zhou, 2018) offered a more adaptable new slicing strategy for the burgeoning collaborative 3D printing platform characterised by several mobile MEX machines. McPherson suggests a chunk-based slicing system to divide the print job into chunks so that various mobile printers can print parts simultaneously without interfering with one another. As the size of the print and the number of mobile MEX machines rises, the advantages of this approach show that a collaborative printing process avoids the significant temperature differential and related internal stress (Poudel et al., 2020).

However, none of these contributions gives a straightforward, repeatable and low-budget method for planning the trajectories of a gantry-based system equipped with a multitude of extrusion heads, being able to avoid collisions among the extrusion head and using today's slicing tools (such as (Ultimaker Cura)), whose output is the G-code file. The G-code data tells a machine controller where the motors should go, how quickly they should move, and what path they should take (Di Angelo et al., 2020; Rais et al., 2021).

The state-of-the-art analysis reveals a technological gap to be filled. Thus, this contribution proposes an innovative and reproducible methodology to plan the toolpath of a gantry-based multi-head MEX framework using only the information contained in a traditional G-code file, typical of a dependent multi-head FDM machine. The manufacturing of a machine's prototype will be carried out in the future.

In particular, this study would ideally take a step ahead in developing an operational gantry-based collaborative large-scale MEX framework by managing the toolpath of the multitude of extrusion heads. The toolpath file is converted into a new format optimised for a collaborative and independent multi-head MEX machine. Pseudo-codes herein describe the innovative methodology, and the outcomes of virtual tests on two real-life large-scale components are included. At first, the 3D model of a jet engine bracket, a benchmark object by General Electric Company (General Electric Company, 2013), is used to validate the methodology with a two-extruder architecture. Then, an entire UAV structure has been considered to demonstrate the consistent building time reduction thanks to the employment of a collaborative MEX four-head philosophy. The proposed methodology will be physically tested on a gantry-based multi-head system in the future once the hardware design and production are completed.

This paper is organised as follows: Section 2 addresses a brief review of existing AM collaborative frameworks available in the literature; Section 3 covers the description of the innovative methodology to decompose the traditional G-code into a multitude of toolpaths, avoiding dangerous collisions; Section 4 contains both case studies that have been used to test the proposed approach along with the discussion of the results. Section 5 highlights some conclusions and future developments.

Collaborative material extrusion architectures with independent extruders: the state of the art

The anticipated trilemma leads to the need for practical solutions, such as the introduction of MEX designs with multiple extrusion heads: as a consequence, a modification in both machines and process planning to manage print head synchronisation must be adopted. Thanks to this kind of AM machine, it could be feasible to create large-scale components without sacrificing quality and reducing production times. A few multi-head AM machine concepts are known in the literature and are discussed in the following.

A patent has been filed for a modular machine with many print heads coupled statically in the same gantry for duplication purposes (Fontaine, 2016), comparable to the IDEX (Independent Dual Extruder System), which enables duplication and mirror printing modes capable of producing two parts at once (BCN3D). However, this solution cannot be applied to a single significant component where the cloud of heads works simultaneously.

The source (Wachsmuth, 2008) details the first effort to create a collaborative AM machine with a multi-head architecture that may print many components or work on a specific region of a larger part. However, the planned hardware and framework solutions are complex and challenging to replicate, and AM technology has advanced fast since 2008.

Another notable project is (M. Leite et al., 2018), in which the authors suggest a modular gantry-based system wherein each head may partially superimpose the regions covered by the others, resulting in an ideal merging of distinct subparts. Several gantries with varied heads and separate print beds are intended to allow different-sized parts to be printed simultaneously.

A relevant family of projects (McPherson and Zhou, 2018; Poudel et al., 2020; Zhang et al., 2018) tries to solve the trilemma issue (high quality, large-scale components and low build time) using a cooperative farm of mobile MEX machines. In Zhang's research, highly sophisticated and expensive equipment, such as cameras, are installed in the mobile MEX machines to guide them into the workspace and avoid problematic accidents using collision avoidance optimisation techniques (Claes and Tuyls, 2018). However, the initial investment to purchase many arm-based MEX systems is far higher than a gantry-based multi-head cartesian FDM machine. As a last concern, the attention is not focussed on an optimal bonding strategy between subsections to avoid weak spots inside the manufactured structures. On the other hand, McPherson presents an attractive strategy to avoid collision between arm-based systems using an optimised chunk-based subdivision of the large-scale component to be manufactured without the need for expensive equipment. Each chunk can be manufactured only by a single machine. Through optimised manufacturing planning, based on the design of a directed dependency tree, the authors prevent dangerous collisions by printing neighbour chunks in two different temporal stages by different machines.

Last but not least, it is worth mentioning Autodesk's Escher Project, a sophisticated control technique for gantry-based FDM technique that can coordinate the motions of a group of extruder heads spread out in one direction. The newly created control software allows them to operate together to print large-scale parts as a single component, preventing dangerous collisions and speeding up the manufacturing process considerably: Titan 3D Robotics, the firm that created the Cronus (Robotics) machine, integrated and utilised this type of technology. To the best of the author's knowledge, how it works is not widely documented in the literature, and it appears to be a dormant project since no publication of further advances has been made since 2017.

From this quick summary of available resources dealing with collaborative and large-scale material extrusion frameworks, the scientific community is concentrating on finding intriguing solutions to the above-described conflict of objectives. However, in such a complicated system where numerous extrusion heads operate in proximity to each other simultaneously, there is still a gap to bridge in tool path management for gantry-based frameworks to avoid dangerous collisions.

Wang et al. tried to answer this open challenge (Wang et al., 2017); their approach mainly focuses on the optimal area subdivision meanwhile guaranteeing collision avoidance. Their method splits each layer into conflicting and non-conflicting sections; the extruders may clash in a conflicting area that should be printed individually by a single extruder, while the non-conflicting areas can be printed in parallel by all the extruders. However, their technique is solely mathematical and has not been tested; also, no mention is made of how the new G-code is formed or if it can be generated using regular slicing tools.

Because of the still unresolved issue in the literature, this research aims to propose a methodology to efficiently complete the toolpath planning task knowing only the information provided in the G-code file and avoiding undesired collisions. Thus, the authors illustrate a reproducible and never-explored strategy to partition a G-code produced by the traditional slicing software into n distinct G-codes, each for every extruder. The proposed methodology avoids collisions between neighbour extruders by simply checking the relative distances of neighbour heads during the manufacturing paths, without the need for expensive sensors and using some pause commands to temporally hold the extruder's position when close to a neighbour one. Thanks to its implementation-oriented approach, this research would ideally take a step ahead in developing an operational collaborative large-scale MEX framework. The following section contains a detailed description of the methodology and some ideas for a possible conceptual design of the AM machine.

G-code partitioning methodology

The proposed approach for G-code partitioning is thoroughly examined in this section, including flowcharts and pseudo-codes. This research can be seen as the natural continuation of the project described in (Bacciaglia et al., 2022); thus, the reader could refer to it for additional information not included in this manuscript.

The proposed methodology can be compatible with a collaborative gantry-based large-scale FDM architecture with n independent extrusion heads and n working areas circumscribed by common working regions where neighbour extruders could work. According to an ideal printing bed visible in Fig. 1, each extruder T_i (with $i \in [0, n-1]$) works on a region coloured in light blue, which is limited in size.

The common working region dimension is defined as equal to the width of the subdivision shape sketch described in the following. These common working regions, visible in dark blue in Fig. 1, where neighbour extruders could work simultaneously on the same component, allow the optimal bonding regions between subparts of the large-scale item.

The description of the overall methodology phases is graphically supported by the flow chart included in Fig. 2 and herein described in detail.

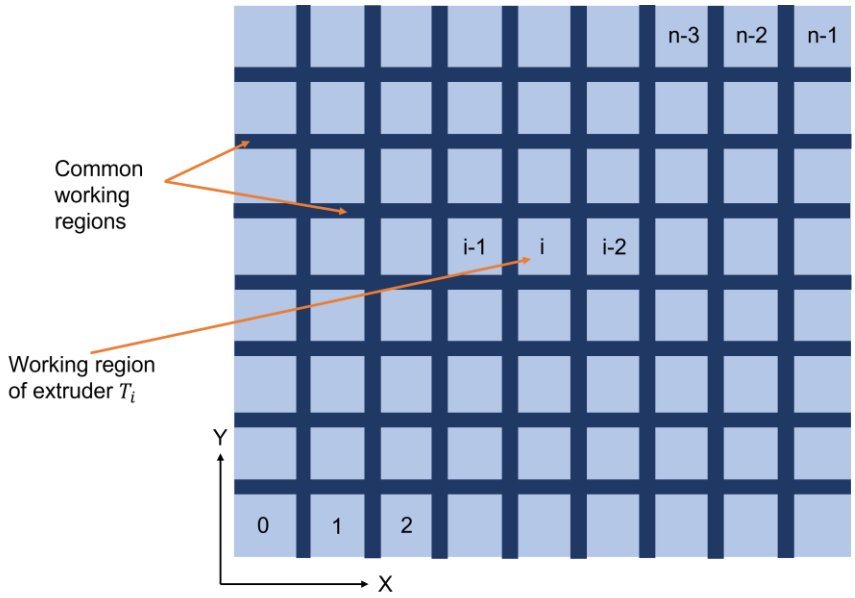

Fig. 1 Schematic view of a collaborative FDM architecture bed subdivision with n printing heads: in light blue each working area; in dark blue, the common working regions that delimitate all the working areas.

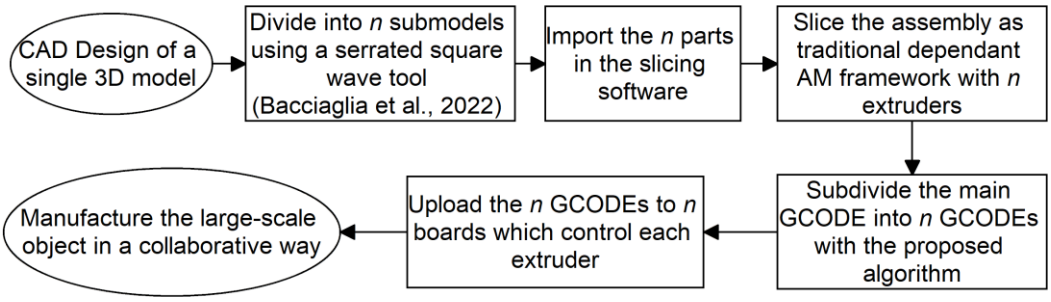

Fig. 2 Graphical flowchart of the proposed methodology to subdivide a traditional G-code and adapt to a collaborative FDM architecture.

As the first step, the designer is asked to design the 3D model of the large-scale object he/she would like to manufacture using standard CAD software tools. The geometry of the digital model of a large-scale component is then automatically preprocessed using the novel technique described in (Bacciaglia et al., 2022) before importing it into a slicing programme. This approach is used to partition a 3D model into manageable portions using Boolean operations in CAD software to produce n submodels making the use of a serrated tool to create an overlapping region with a square waveform along the growth direction (Fig. 3) that will be manufactured simultaneously by two neighbour extruders in the joint working regions.

As shown in Fig. 3, the overlap region length along the XY plane gives the width of the common working area visible in Fig. 1 and can be adjusted by the user depending on the bonding strength he/she would like to achieve. Further analyses on the dependency of the square wave shape and the bonding action will be investigated in future research.

Compared to the chunk-based approach proposed in (McPherson and Zhou, 2018), where the bonding surface consists only in an inclined plane, thus creating a weak region along a plane, in the proposed approach, the joint is based on a squared-wave shape with a significant overlap region in the in-plane direction. Thus, an optimal bonding between subparts is obtained without manual glueing post-process. Indeed the bonding is achieved using the traditional intralayer bonding action typical of MEX processes. Moreover, compared to the McPherson approach, the square-wave subdivision described in (Bacciaglia et al., 2022) can be applied to complex freeform shapes without affecting the external surface continuity since it acts only on the inner volume of the large-scale component and leaves untouched the

external boundaries. As a last point, to the best of the authors' knowledge, the chunk-based approach by McPherson has not been tested on freeform shapes typical of aerospace and automotive structures.

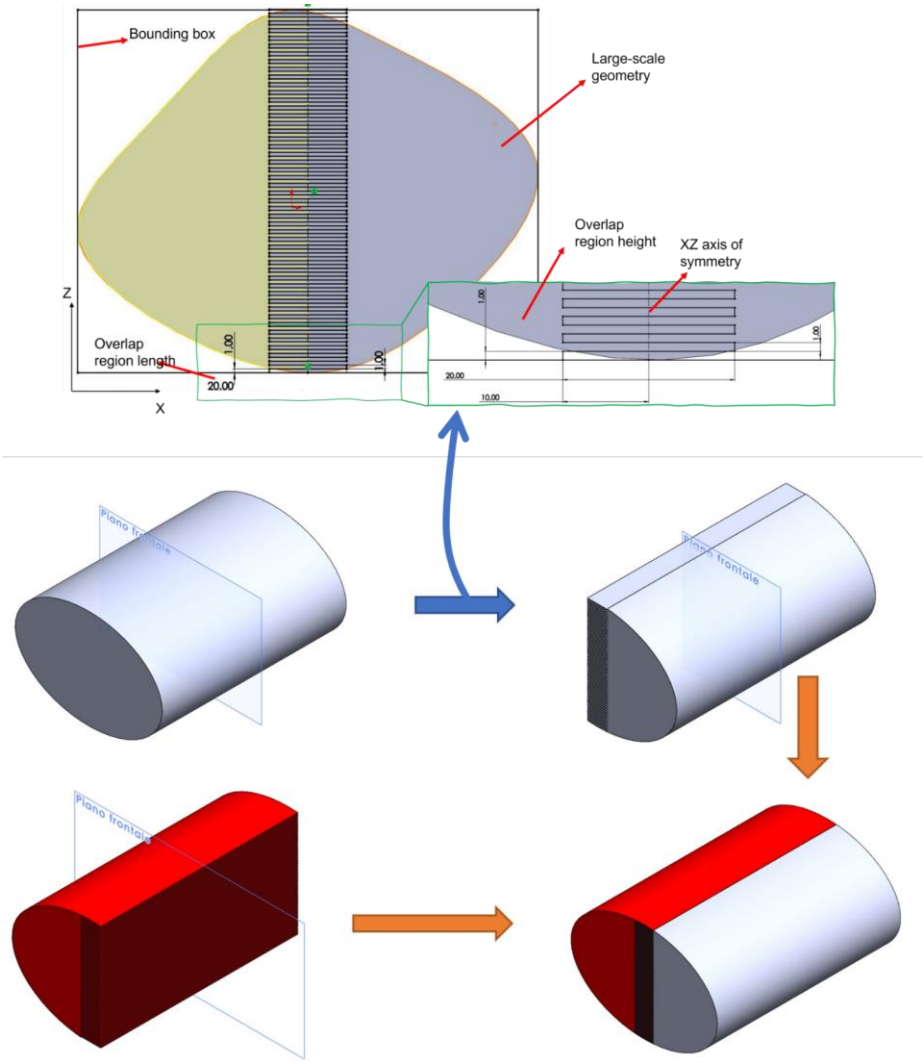

Fig. 3 The subdivision methodology with $n = 2$ described in (Bacciaglia et al., 2022) applied on a freeform surface.

The squared-wave shape is required to avoid weak spots inside the final object without affecting the external surfaces (Duran et al., 2015). At the end of the digital model manipulation, the CAD software should output n STL files (Hiller and Lipson, 2009), each one for a single extrusion head, which can be imported and assembled in typical slicing software, such as Cura Ultimaker.

In the next phase, the user slices the large-scale component made as an assembly, assigning a single extruder to a single part as a dependent FDM architecture (Butt et al., 2018). In addition, the required 3D printing settings, such as layer height, extrusion head's temperature, extrusion velocity, per cent of infill, and so on, should be selected. The slicing software outputs a unique G-code file containing all the required printing settings and the extruders' toolpath. In traditional multi-head dependant architectures, a section of instructions about a single extruder can be recognised by G-code lines where the T_i command alone is used to select the i-th extruder. The reader is referred to (Krishnanand et al., 2021) for further information on the standard G-code file structure.

Then, the unique G-code file is scanned, and a set of operations are carried out to provide n separate files containing the toolpath information regarding a single subpart of the large-scale assembly that automatically prevents collisions between the cooperative extruders. The algorithm has been implemented in MATLAB and will be detailed in the following paragraph. Indeed, using the proposed algorithm, it is feasible to split a single G-code into n separate files, one for each extrusion head, and use

all n heads simultaneously, avoiding harmful collisions and optimising printing times. A typical scenario where this approach could be used is a large-scale MEX/FDM architecture with n motherboards, one for each extruder, which controls the extruder's motion in its dedicated working area. The main motherboard that controls the initial instant of time and the overall process's simultaneity is also necessary. However, the hardware design of the architecture is beyond the research's scope, and this topic will be addressed in future developments. It is worth noting that the design and production of a gantry-based multi-head MEX framework requires a preliminary validation phase which is presented in this paper. Therefore, before verifying the path planning algorithm on a real AM machine, the authors developed a simulation environment in MATLAB to test the toolpath over different layers using the sliced results as input, which will be visible in the following.

After the computation of the n toolpath files, the output may then be transmitted to the independent and collaborative multi-head FDM architecture herein introduced, where the manufacturing process can start to manufacture a large-scale component efficiently.

Extrusion path planning methodology

This section will explain the authors' strategy for separating and making independent the single G-code file produced by common slicing tools for dependent multi-head extrusion and, thus, obtaining n separate G-code files. Tables I and II provide the pseudo-codes of the methodology to make it reproducible and understandable at the end of this subsection.

Before the algorithm is launched, the user must specify the overall extrusion head size (e_{dim}), while the layer height, the extruder's velocities and the temperatures of the multitude of heads are automatically recognised by reading the unique G-code file from the slicing software. The G0 and G1 commands are given special consideration, which enable the printer to execute the linear move instruction and allow the extruder to move in the desired direction. These commands are correlated to specific extruder speeds chosen in the slicing software, which in the following will be mentioned respectively as V0 and V1. The G1 command permits the printer to extrude the raw material during the linear move, whereas the G0 instruction is for movements without extrusion (Faria et al., 2020).

The algorithm starts by scanning the G-code file and saves the total number of layers of the overall project in n_{layers}. Moreover, a six-column matrix M is created to gather the instructions based on the kind of command (G0, G1 or extruder change, spatial coordinates, extrusion quantity, feed rate change), with the typical structure shown in Fig. 4.

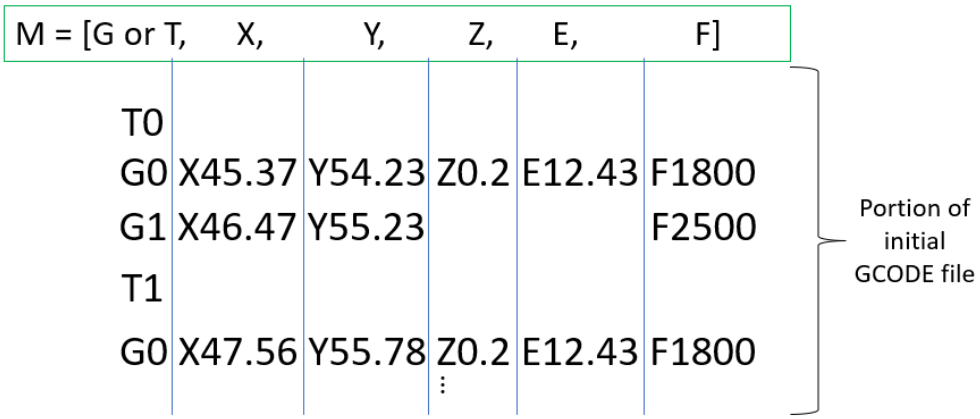

Fig. 4 Information extracted from the G-code is recognised and saved in the proper column of the *M* matrix.

In the following, it is feasible to segment the obtained matrix into n matrices called M_i, one for each extruder, by scanning M and looking for lines having the instruction of the activation of a single extrusion head. In particular, the existence of a line with a single instruction made by $T_0, T_1, T_2, \dots, T_i, \dots, T_{n-1}$, means a change of the active extruder. Such extruder recognition and subdivision can be graphically appreciated in Fig. 5, 6 and 7, which contain the path reconstruction in

MATLAB of a sphere chunk for a $n = 2x2 = 4$ machine. A different colour is assigned to each extruder; knowing the M_i matrices that are produced by the proposed methodology, it is possible to reconstruct the trajectories of each extrusion head. A new line is also inserted at the start of each matrix M_i to establish the location of each extrusion head to its home: for example, the extruders will be placed at the four borders of the printing bed for a $n = 2x2 = 4$ architecture.

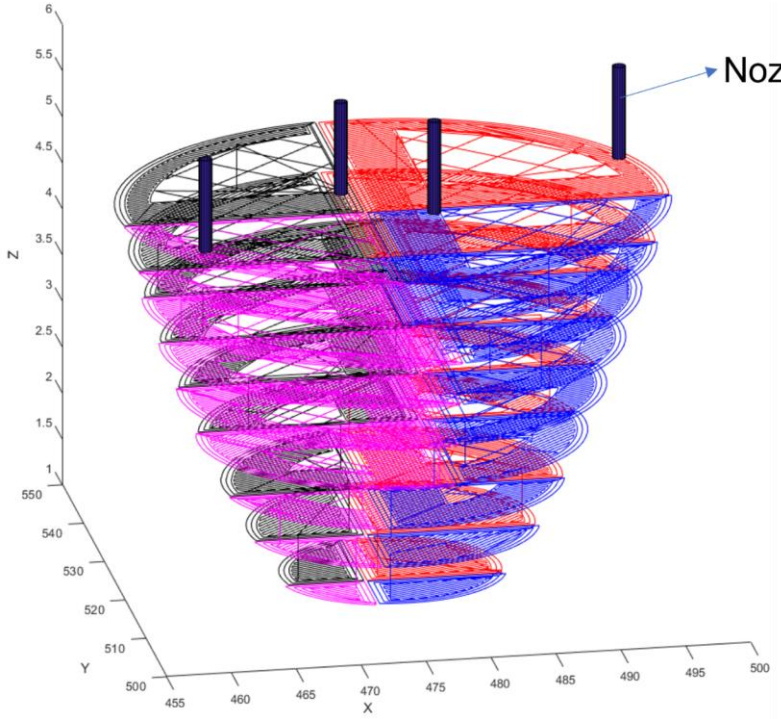

Fig. 5 Reconstruction of extruders' trajectory in MATLAB of a chunk of a sphere divided into $n = 2x2 = 4$ parts.

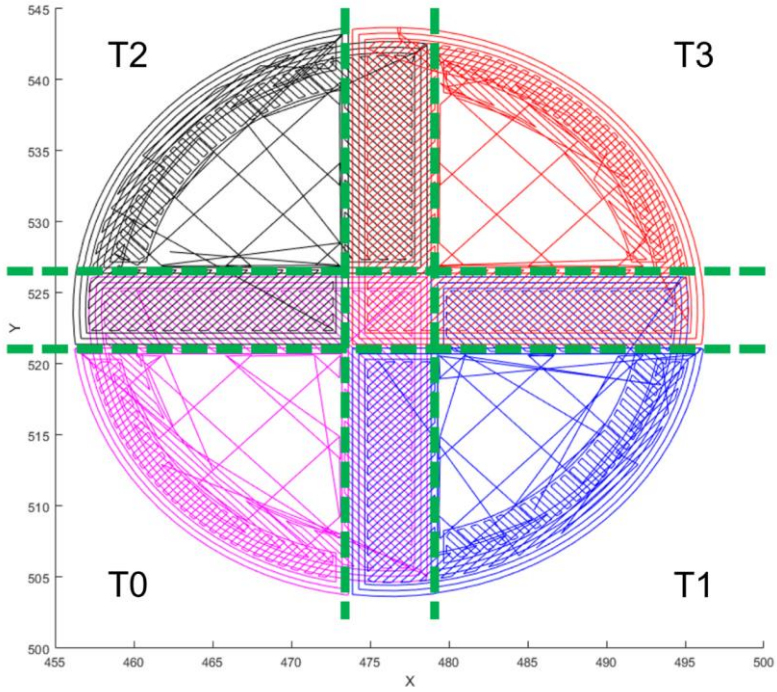

Fig. 6 Detailed view of the reconstruction of extruders' trajectory in MATLAB from the top of a chunk of a sphere for a $n = 4$ architecture; the common working area is highlighted in a green dot line.

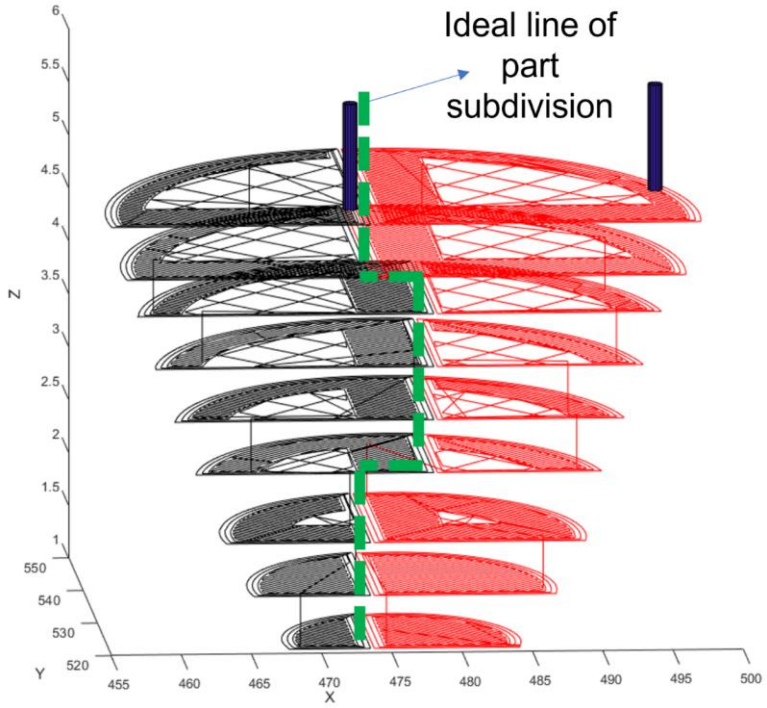

Fig. 7 Reconstruction of extruders' trajectory in MATLAB: a detailed view of the square wave line applied to half of the sphere.

Then, for each line of the M_i matrix, the algorithm evaluates the time the AM framework needs to carry out the tool's instruction therein contained, knowing the operating speeds set in the slicing software. Several cases can happen, depending on the type of two consecutive instructions contained in the M_i matrix:

- If the j-th line contains a G0 instruction and the extrusion head displacement from j-1 to j is greater than 0 (meaning that a displacement occurs), then the time needed is equal to the ratio of the displacement over V0;
- Otherwise, if the j-th line contains a G1 instruction and the extrusion head displacement from j-1 to j is greater than 0, then the time needed is equal to the ratio of the displacement over V1;
- Otherwise, if moving from j-1 to j line, the Z coordinate increases, then it means that there is a layer change, and the time is equal to the ratio of the Z increment over the Z-axis velocity;
- Otherwise, if moving from j-1 to j line, the F feed rate value changes, then it means that there is no displacement in the XY or Z direction, and the time spent is concise (a 1/1000 of a second could be a good pre-set value);
- Otherwise, a retraction instruction happens, and the time spent equals the extrusion increment/decrease over the extrusion velocity.

By implementing these cases, for each line of M_i the algorithm is capable of evaluating the time required to complete a single instruction to complete the manufacturing of a single subpart; the instruction time is saved in a new column of M_i. Furthermore, for each j line of the M_i matrix, the cumulative time t_{ci} is calculated and saved by adding the total time required to complete the instructions of the preceding j-1 lines. Beginning with time 0, which corresponds to the start of the operations, the cumulative time is crucial for understanding the relative distances between the n extruders at any given instant of time during the AM process.

All the calculated timings are used to automatically determine the relative distances between the extruders during the manufacturing of a single layer of the overall project. Thus, the algorithm takes a chunk of the n G-codes related to a single layer of the sliced large-scale object for each i extruder and saves it in a temporary variable $temp_i$. The suggested approach is intended for FDM/MEX systems with independent heads in shared gantries. As a result, all extruders should wait for the slowest head to

complete a single layer before starting a new one in an FDM architecture. It would be pointless to verify the extruders' relative distances across multiple layer heights because this condition cannot be physically satisfied by the hardware setup of a collaborative multi-head FDM framework philosophy described in this research.

A static priority strategy for the extruders has been implemented in the initial version of the proposed methodology. Using a cascade process, the static priority strategy analyses the spatial coordinates relative to two neighbouring extruders, where one gets priority over the other. In other words, the extruder T_0 has always precedence over all other extruders at all times; then T_1 takes precedence over the other extruders, and so on. To avoid penalising a single extruder and slowing down the whole process, each extruder should operate on the same area for each layer. Positioning the large-scale assembly in the centre of the printing bed, with the object's centre of gravity aligned with the centre of the plate, could be a straightforward strategy for satisfying this criterion. A dynamic priority strategy will be investigated in the future to make the algorithm adaptable to all possible situations.

To describe in depth the static priority strategy implemented in this research, the extruders T_0 and T_1 are chosen as a reference, as well as their temporary matrices $temp_0$ and $temp_1$. The implemented algorithm can determine if a probable collision may occur during the heads' movements at a certain point in the manufacturing process within the same project layer. Scrolling $temp_1$, at each instant saved in tc_1, the current and prior locations of the extruder T_1 are recorded in two new variables called $S_1$and S_{1old}. Similarly, the location of T_0 is recorded in S_0 and S_{0old}, at the same time as the cumulative time.

Then, using the implemented function called *check for crash*, whose pseudo-code can be found in Table II, it is possible to estimate if a dangerous crash may occur due to a head collision. Thus, it is feasible to identify if, during the movement from S_{1old} to S_1, the extruder 1 may collide with extruder 0, which, in the meantime, is moving from S_{0old} to S_0. These routes are split into a number m of points (100 in this research) to make the inspection accurate and redundant, as extrusion collisions are a hazardous situation that must be avoided at all costs. This check is performed using the extruders' overall sizes (e_{dim}) and the relative distance between the two extruders throughout their movement. If the relative position mathematical norm is less than the extruder's dimensions, a crash may occur, and in this case, the function returns a positive value.

In the case of possible crashes, extruder 1 is slowed down. In particular, before the new instruction contained in $temp_1$ that moves extruder 1 from S_{1old} to S_1, a pause is inserted, exploiting the G4 command available in Marlin. In particular, the Dwell command is a good choice for delaying the execution of a single G-code instruction, which may be done by specifying a millisecond period, such as G4 P2000, for a 2-second delay. In this research, the required timeframe is evaluated as the time needed for the non-priority extruder to leave enough space for the movement of the priority one by using the lowest velocity between V0 and V1.

For each layer of the project, this check is done for each instruction in $temp_1$. On the one hand, as shown in Fig. 8 with a G-code chunk, the new G-code for the extrusion head T_1 is enhanced with new pause lines when required. The G-code of the priority head T_0, on the other hand, it has not altered since its original form because it has the highest priority on the other heads.

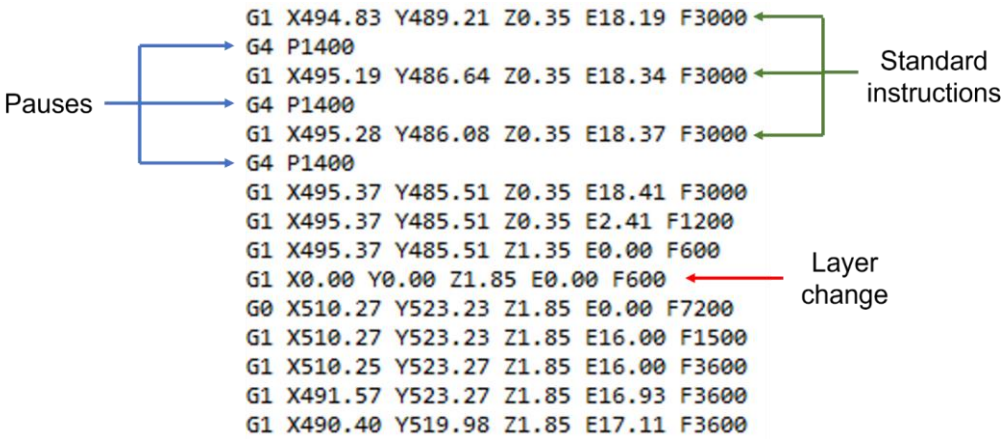

Fig. 8 G-code chunk after applying the proposed methodology with the inserted pauses highlighted with blue arrows.

By comparing the information of the extruder 2 contained in $temp_2$ with the information of extruders T_0 and T_1 (which have priority above T_2), the suggested technique checks for any collisions of

T_2 with priority extruders. With new pauses, if required, the procedure enriches the new G-code, referred to as T_2. In the following, the technique proceeds in a cascade until the final extruder is checked. A filter is implemented to check possible collisions only between neighbour extruders, thus avoiding unnecessary computations between distant heads. Fig. 9 shows a flowchart of the cascade process for the more straightforward design with $n = 2x2 = 4$.

After the toolpath file subdivision, the n G-code files can be sent to the independent and collaborative multi-head FDM architecture, where the manufacturing process can begin. In order to understand if the bonding strategy could work properly, a 3D digital model of a manufactured item has been designed in a CAD package to imitate the inner bonding structure that can be achieved using the proposed methodology. Fig. 10 shows the reproduction of a chunk of the sphere manufactured with a traditional FDM machine (Artillery Sidewinder X1) that attempts to imitate the possible final component manufactured through the collaborative framework, with the stitching between subparts that could be well integrated inside the overall object thanks to the proposed approach. Indeed, the bonding between subparts is included in the infill of the object; thus, the large-scale component's external quality does not suffer from defects. The test demonstrates that a strong bonding can be obtained, and part chunks do not separate. It is worth noting that the code herein developed has been conceived so that each zone of the component to print is assigned to a single printing head.

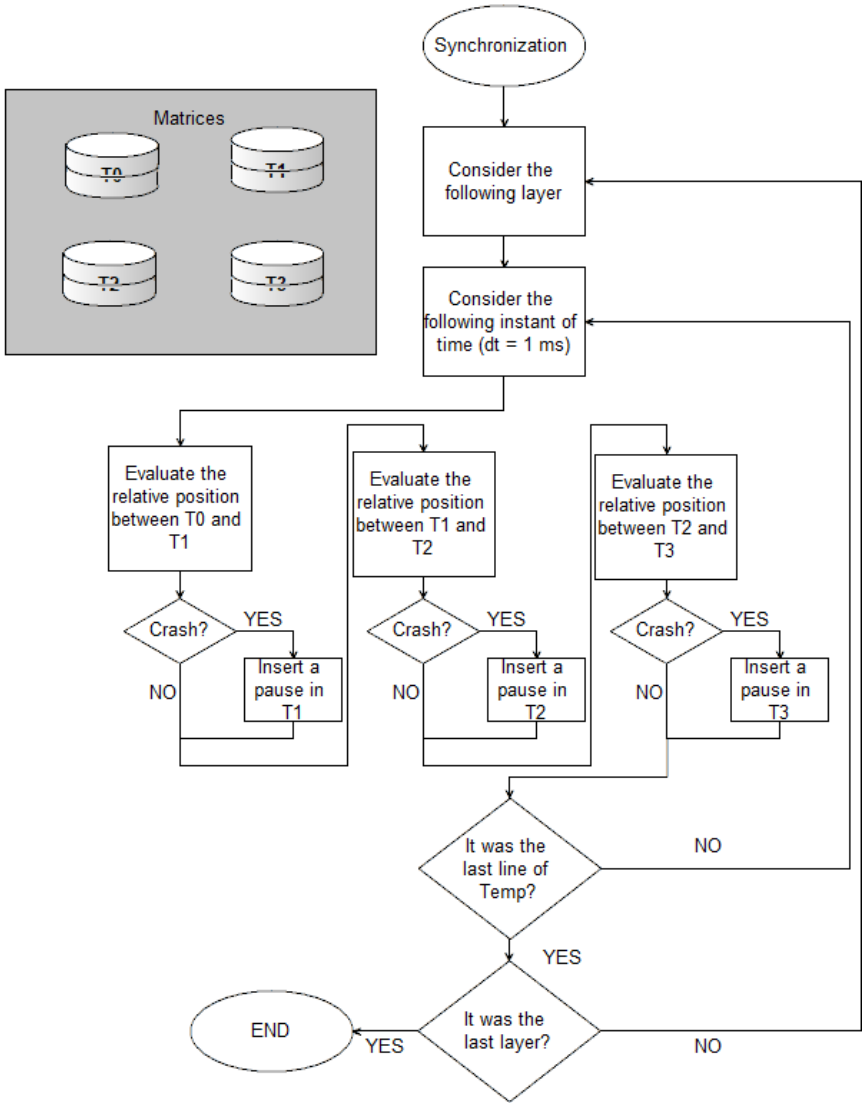

Fig. 9 Static priority concept represented in a flowchart for an FDM architecture with $n = 2x2 = 4$ extrusion heads.

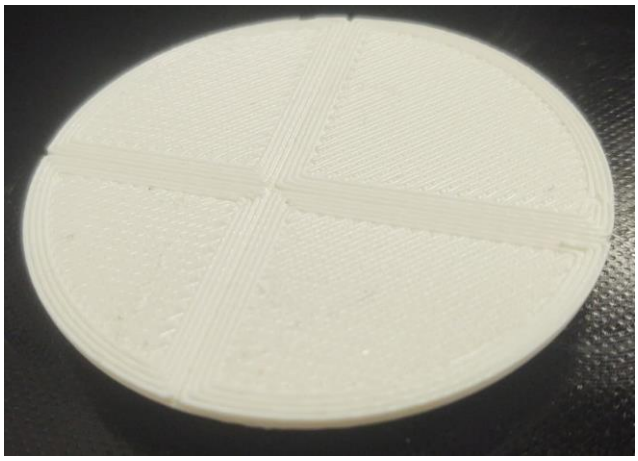

Fig. 10 A 3D printed model of a portion of a sphere: the CAD file has been modified to imitate the shape subdivision proposed in (Bacciaglia et al., 2022); the stitchings are visible inside the final object.

A possible design for a four heads machine which could implement the strategy presented in this paper is included in Fig. 11. It is worth noting that the printing heads must be mounted on curved brackets: this solution allows a printing zone where both left and right heads can operate, and there is no collision within the trusses supporting the printing heads.

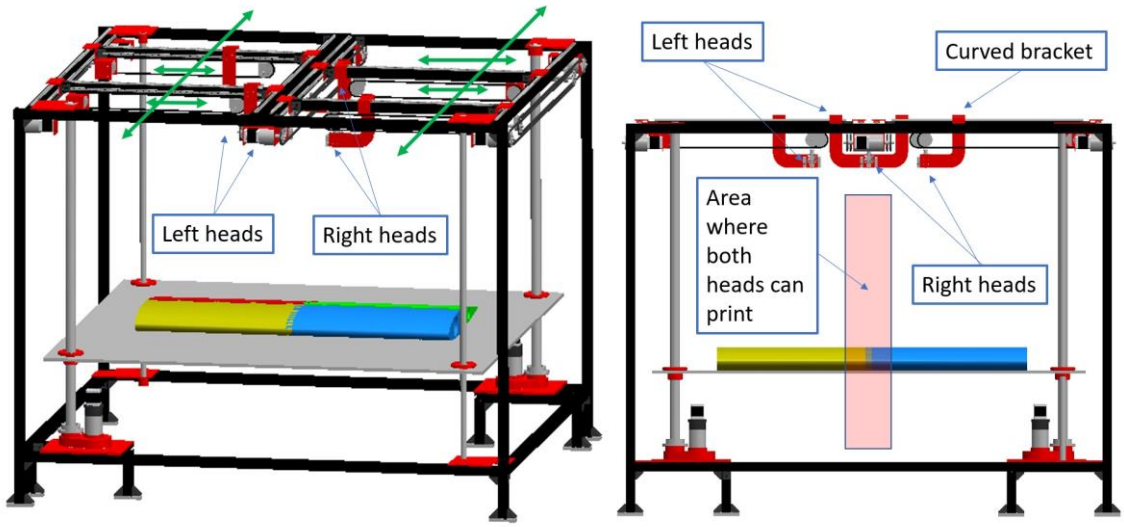

Fig. 11 Conceptual CAD model of a MEX machine with four independent printing heads.

Table I. The pseudo-code of the overall methodology.

GCODE subdivision

```
Input e_dim, V1, V0, Ve and Vz
Home extruders
textscan(GCODE file)
M ← [G or T, X, Y, Z, E, F]
for i ϵ M do
   find (T0, T1, T2, …, Ti) in M
   save portion relative to Ti in Mi
end
n_layers = extract('LAYER COUNT')
from GCODE
for i=0:n-1
   Mi = [home Ti; Mi] %initialize with
      extruder home
end
for  i=0:n-1 do
   for j ϵ length(Mi)
      disp ← sqrt((Xj-1-Xj)^2+(Yj-1-Yj)^2)
         %extruder displacement
      if Mi(j,1)==0 && disp>0
         t=disp/V0 %G0 instruction
      elseif Mi(j,1)==1 && disp>0
         t=disp/V1 %G1 instruction
      elseif zj > zj-1
         t=(zj-zj-1)/Vz %layer change
      elseif Fj ≠ Fj-1
         t=0.001 %feed rate change
      else %retraction
         t=(Ej-Ej-1)/Ve
      end
      Mi(j,7)=t
   end
end
tci=0 % initialise cumulative time
for i=0:n-1
   for k ϵ length(Mi)
      tci=tci+Mi(k,7) %evaluate the
      cumulative time
      Mi(k,8)=tci
   end
end
newgcodei=[]
for j=1:n_layers
   for i=0:n-1
      tempi ← extract portion of Mi ϵ to layer j
   end
   newi=[]
   % check position of M1 compared to M0
   for k ϵ length(temp1)
      tcj=temp1(k,9)
      S1 ← position of M1 at tcj
      S1old ← position of M1 at tcj-0.01
      S0 ← position of M0 at tcj
      S0old ← position of M0 at tcj-0.01
      time=linspace(tcj-0.01,tcj,100)
      output=check_for_crash(S0,S0old,S1,S1old,time)
      if sum(output)>0
         wait=(e_dim/min(V0,V1))
         new1 ← add a P4 command before
               instruction temp1(k,:) of duration
               wait
      else
         new1←temp1(k,:)
      end
   % check position of M2 compared to M0 and
      M1

   %check position of M3 compared to M0, M1
      and M2
   end
end
newgcodei=[newgcodei; newi]
for i=0:n-1
   write gcode file of newgcodei
end
```

Table II. The pseudo-code of the *check for crash* function.

Check for crashes

```
Input S0,S0old,S1,S1old,time
for k ϵ length(time)
   Va_vers=(S0old-S0)/norm(S0old-S0)
   Vb_vers=(S1old-S1)/norm(S1old-S1)
%depending on G0 or G1 instruction, Va and Vb =
      V0 or V1
   Xa=Xa_old+Va*Va_vers*(time(k))
   Xb=Xb_old+Vb*Vb_vers*(time(k))
   if norm(Xa-Xb)<e_dim
      crash(k)=1
   else
      crash(k)=0
   end
end
output crash
```

Case studies

This section provides two case studies to show the methodology's applicability and performance in a real industrial scenario with a large-scale component manufactured with MEX.

GE bracket

In 2013, General Electric Company organised a challenge to design a jet engine bracket using a Topology Optimisation (Sigmund and Maute, 2013) approach (General Electric Company, 2013). The authors selected this geometry due to the complex freeform obtained by the optimiser to demonstrate that the proposed approach can be applied to real-life components. The optimised bracket has a bounding volume box of 535mm x 324mm x 187mm. The 3D model has been divided into two subparts (Fig. 12) using the (Bacciaglia et al., 2022) approach, and assign each part to different extrusion heads (T_0 and T_1).

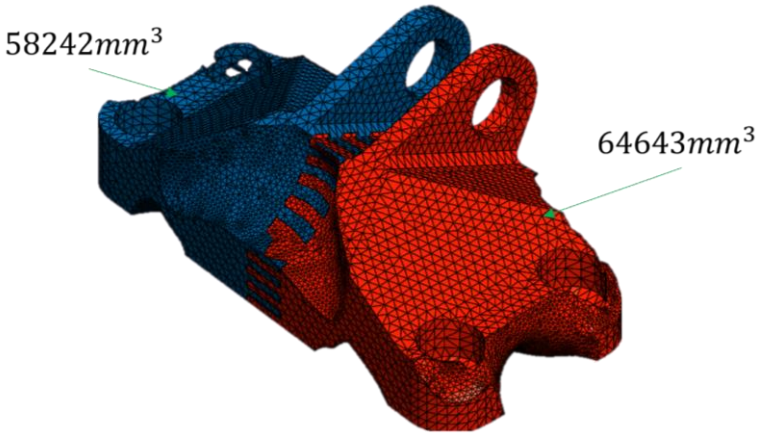

Fig. 12 GE jet engine bracket subdivision in two parts for an FDM architecture with $n = 2x1 = 2$ extrusion heads.

The GE jet engine bracket model could be manufactured using a traditional FDM machine with an adequate building size in 114 hours, utilising conventional single extruder settings such as 10% infill, 0.3mm layer height, and 60 mm/s extrusion velocity. However, the estimated manufacturing time is hardly compatible with modern time-to-market requirements. Thus, a two-head MEX framework has been chosen to understand the impact of the approach proposed in this manuscript and to verify if a consistent reduction of the manufacturing time can be achieved.

The two STL files were saved, imported into Cura Ultimaker and merged. Standard slicing settings were provided, and a unique G-code of 8 MB and 103,163 lines was built. This file provided the starting point for the technique herein presented to plan the toolpath for a collaborative FDM framework.

The algorithm was tested on a workstation with 32 GB of RAM and a 3.50 GHz Intel Zeon CPU. It took 17 minutes to read, separate, inspect for accidents, and rewrite $n = 2$ updated and independent G-code files. These files, which have 46,200 command lines and a size of 3.5 MB, can be considered ready for upload into a cooperative and autonomous multi-head FDM architecture.

With a collaborative architecture, predicting the new production timing is feasible by knowing the cumulative time of the slowest extruder. As can be seen in Table III, it could be possible to manufacture the bracket's slowest element in 79.2 hours if the best combination is chosen. Thus, a consistent reduction of the manufacturing time by over 30% can be achieved using a collaborative and multi-extruder FDM framework.

Table III. Study of both priority combinations vs total manufacturing time; in light green, the optimal solution

Static priority combinations	T_0 time [h]	T_1 time [h]	Tot time [h]
$T_0; T_1$	79.2	61.5	79.2
$T_1; T_0$	58.3	86.9	86.9

The path reconstruction in MATLAB of some layers of the GE bracket for an $n = 2x1 = 2$ MEX framework, where each extruder is given a unique colour, is shown in Fig. 13.

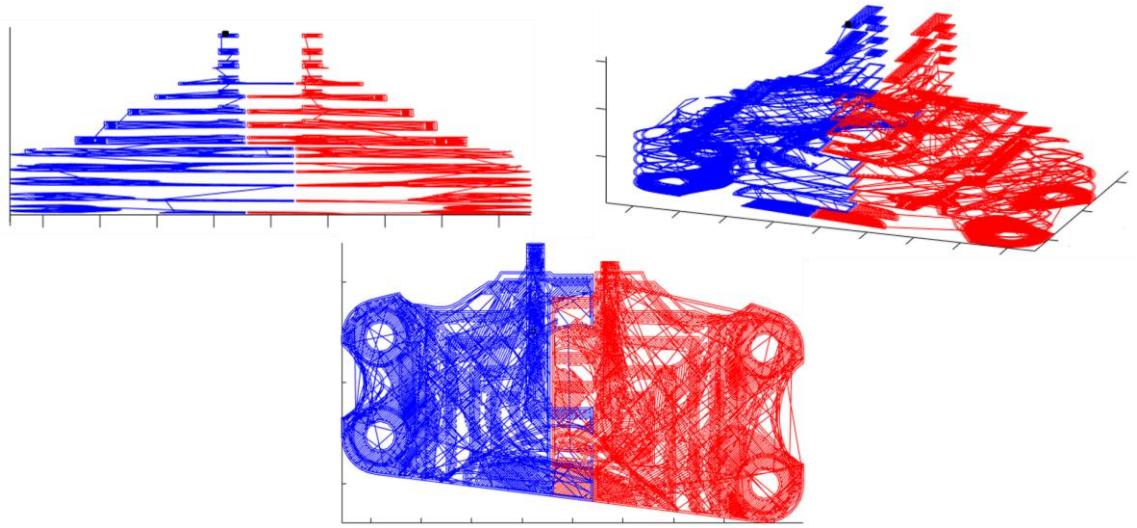

Fig. 13 Reconstruction of extruders' trajectory of some layers of the GE jet engine bracket in MATLAB from different views.

Flying-wing UAV

The AM manufacture of a fully dense and real-scale mockup model for wind tunnel tests of a 1-meter span flying-wing UAV (Fig. 14) has been chosen as another suitable scenario for testing the proposed method with more than two extruders. Nevertheless, the method could also be applied to the complete model made of thin-shell parts and empty areas.

Using conventional slicing settings such as 20% infill, 0.2mm layer height, and 50 mm/s extrusion velocity, the real-scale wind tunnel UAV model could be built in 75,78 hours using a traditional FDM machine with appropriate construction volume using a traditional single extruder setup. However, this production time is undesirable in an industrial environment, and a quicker manufacturing approach should be used. Indeed, in a real industrial scenario, the design-to-manufacturing cycle has strict deadlines that the companies should face. Overcoming the slow production rate of traditional FDM/MEX frameworks could represent an advantage for applying the proposed methodology; a traditional and single G-code is subdivided into a multitude of toolpath data files for a multitude of independent extrusion heads to manufacture a large-scale model within a collaborative FDM framework.

According to the proposed approach, the UAV was first segmented for simplicity into four sections, using a serrated square wave tool, with particular attention to the subdivision phase into sub-parts: the bonding between subparts was placed strategically to obtain four components of the same volume (Fig. 15).

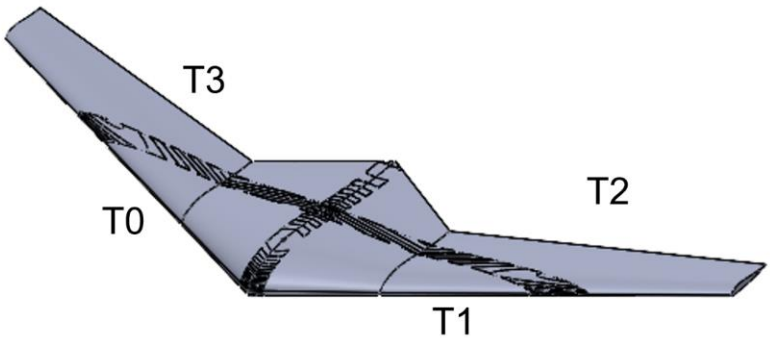

Fig. 14 Flying-wing UAV subdivision into four parts for an FDM architecture with $n = 2x2 = 4$ extrusion heads.

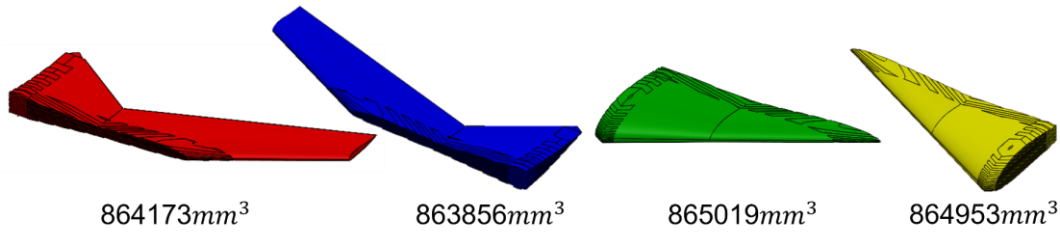

Fig. 15 The four parts of the flying-wing UAV have been isolated, and the value of part's volume was calculated.

The four STL files were saved, loaded into Cura Ultimaker, and combined. A unique G-code of 27 MB with 91,693 lines was created after specifying standard slicing settings. This file was used as the input data for the proposed methodology to plan the toolpath for a collaborative FDM framework efficiently. Fig. 16 collects the toolpath reconstructed in MATLAB for the UAV case study.

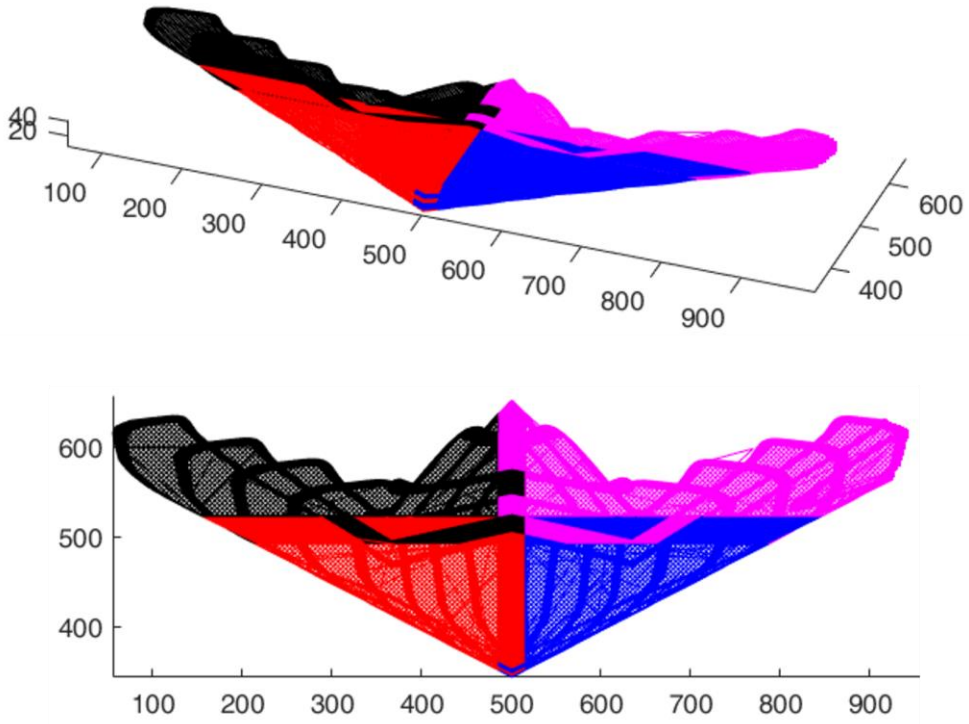

Fig. 16 Reconstruction of extruders' trajectory of some layers of the flying-wing UAV in MATLAB from different views.

Reading, splitting, checking for crashes, and rewriting $n = 4$ updated and independent G-code files took 31 minutes. These data, which have an average weight of 8 MB and 22,291 instruction lines, can be considered ready to be uploaded into a collaborative and independent multi-head FDM architecture, including four control boards and one mainboard dedicated to directing the process's simultaneity.

The slowest subpart of the UAV could be manufactured in 19.3 hours for this specific case study, using the extruders' assignment shown in Fig. 14. Using a collaborative and multi-extruder FDM framework saves 74.5% of the whole manufacturing process time. The increase in manufacturing efficiency is in line with the results available in the literature, such as (Wang et al., 2017), which says that the printing efficiency is not precisely n times when compared to single extruder, but few decimals below. In this particular case study, following the definition provided by Wang, the printing efficiency has been increased by a factor k equal to:

$$k = \frac{75{,}78}{19{,}3} \cong 3{,}93 \qquad (1)$$

This encouraging finding indicates that the main request that prompted the algorithm implementation was adequately answered.

However, it is essential to understand which is the best priority order of extrusion heads that reflects the shortest manufacturing process: at least in this first implementation, the proposed methodology is limited to a static priority strategy. However, using an FDM architecture with 4 extruders, the possible priority combinations are $n = 4! = 24$. Automatised and more efficient optimisation strategies will be studied in future. In particular, using the notation of Fig. 1 applied on a 2x2 FDM architecture, the static priority combinations applied to the flying-wing UAV of Fig. 14 that are contained in the first column of Table IV have been investigated. Analysing all possible combinations took 3 hours and 41 minutes on the same workbench used in other simulations.

Table IV. Study of all priority combinations vs total manufacturing time; in light green, the optimal solution

Static priority combinations	T_0 time [h]	T_1 time [h]	T_2 time [h]	T_3 time [h]	Tot time [h]
$T_0; T_1; T_2; T_3$	13.0	22.4	12.7	17.2	22.4
$T_0; T_3; T_1; T_2$	13.0	12.9	29.8	16.0	29.8
$T_0; T_3; T_2; T_1$	13.0	12.7	24.3	17.2	24.3
$T_0; T_2; T_1; T_3$	13.0	15.4	22.2	17.3	22.2
$T_0; T_1; T_3; T_2$	13.0	22.4	12.9	16.0	22.4
$T_0; T_2; T_3; T_1$	13.0	12.9	16.0	30.6	30.6
$T_1; T_0; T_2; T_3$	15.1	14.9	12.7	20.2	20.2
$T_1; T_0; T_3; T_2$	15.1	22.4	12.9	16.0	22.4
$T_1; T_3; T_0; T_2$	15.1	12.6	25.2	17.2	25.2
$T_1; T_3; T_2; T_0$	15.1	17.3	25.9	30.6	30.6
$T_1; T_2; T_3; T_0$	15.1	16.2	29.8	16.0	29.8
$T_1; T_2; T_0; T_3$	15.1	22.4	12.9	16.0	22.4
$T_2; T_1; T_0; T_3$	12.5	17.7	25.2	17.2	25.2
$T_2; T_1; T_3; T_0$	12.5	22.4	12.9	16.0	22.4
$T_2; T_3; T_1; T_0$	12.5	15.9	24.8	27.0	27.0
$T_2; T_3; T_0; T_1$	12.5	14.9	12.9	19.3	19.3
$T_2; T_0; T_3; T_1$	12.5	23.6	15.9	26.8	26.8
$T_2; T_0; T_1; T_3$	12.5	17.3	24.8	27.0	27.0
$T_3; T_1; T_0; T_2$	12.7	23.6	17.8	16.0	23.6
$T_3; T_1; T_2; T_0$	12.7	17.7	25.2	17.2	25.2
$T_3; T_2; T_1; T_0$	12.7	16.1	29.8	16.0	29.8
$T_3; T_2; T_0; T_1$	12.7	17.7	17.2	26.8	26.8
$T_3; T_0; T_2; T_1$	12.7	12.7	17.2	30.6	30.6
$T_3; T_0; T_1; T_2$	12.7	14.8	12.7	19.8	19.8

From the whole set of results shown in this section, it is possible to conclude that the aim of the proposed methodology is wholly addressed, and encouraging reduction of manufacturing time can be achieved using efficient toolpath management based on a collaborative and independent multi-head FDM architecture. Even a simple toolpath planning that uses a static priority strategy can efficiently speed up the slow production rates of AM for large-scale components. Moreover, the authors believe that dynamic strategies that will be investigated in future works could further improve the efficiency of the approach. Large-scale objects with low infill percentages should be approached cautiously: because of additional outer wall layers (the external walls of the sub-parts) where the stitching is present, a slight weight gain should be expected.

Conclusion and future developments

This study aims to provide an original, trustworthy, and reproducible strategy to plan and subdivide a single toolpath data file generated by today's standard slicing tools and adapt it for a collaborative multi-extruder FDM framework. By employing such architecture, a large-scale component could be efficiently manufactured with fast production rates while maintaining satisfactory quality. Indeed, a conflict of objectives exists in fabricating large-scale components: with standard FDM machines, designers may only minimise two of three parameters between surface roughness, printing

time, and size constraint. High surface quality, decreased printing time, and a considerable printing volume may all be achieved together by transitioning from the traditional single-extrusion head FDM technology to a large-scale collaborative framework where many extruders can work simultaneously on the same component. In this scenario of technological transition, the approach herein described would ideally take a step ahead in developing an operational collaborative large-scale MEX framework by managing the toolpath of the multitude of extrusion heads. Thus, the proposed G-code subdivision technique aims to output n separate files depending on the number of available extrusion heads: this to substantially reduce total printing time.

The approach applies to all independent FDM architectures with multiple heads positioned in the same gantry and free to operate in a constrained rectangular zone circumscribed by a shared working area where all the extruders from neighbouring regions can work. Applying a partial overlap along the different layers makes it possible to better homogenise the stitching across independent subparts and eliminate weak points in the inner structure without affecting the external skin quality.

The head's trajectory data is rearranged using a static priority technique in priority cascade mode. This is done to strategically plan the toolpaths in the joint working regions of neighbour extruders and minimise harmful collisions. Indeed, the system prioritises the highest priority extrusion head above the others in advance, pausing the neighbouring ones. This check is performed by measuring the relative distance between the extruders, which could collide in the production process. If a collision is possible, the algorithm inserts a pause command in the non-priority head, preventing crashes. The n separate G-codes that have been rearranged and produced by the proposed methodology could be ready to be uploaded in a collaborative FDM framework. Such architecture should be designed to have n corresponding control boards, one for each extruder, that monitor the head's movements. Furthermore, a separate board ensures that the procedure runs in parallel.

Two case studies with $n = 2$ and $n = 4$ have been included to describe typical industrial scenarios involving the production of a jet engine bracket and an entire UAV fuselage: the proposed approach increases the printing efficiency by a n factor, in line with the results available in the literature, by rearranging the tool paths. In a real-world scenario, such a figure of merit is immensely enticing, allowing AM enterprises to improve their production volumes: this could be particularly useful in large-scale component manufacture, which has been severely hampered by the limitations of 3D printing up to our days.

Soon, the approach will be enhanced with a dynamic priority strategy that will dynamically adjust the precedence order during the AM process, shortening the manufacturing timings even more. The design and production of the effective collaborative MEX machine will be discussed in future research. Furthermore, particular emphasis will be paid to optimising slicing parameters to achieve the optimum stitching between independent subparts and to minimise filament cross-contamination on the final object during the induced pauses by optimising retraction settings. Additional tests are also required to determine how the design and the thermal issues about the stitching zones' extrusion process, particularly the serrated tool shape, may affect the final body's strength. Lastly, rather than confining the methodology to cartesian machines, applying the same approach to delta machines, which have a radically different system of gantries and handling, could be interesting.

Declarations. This research received no specific grant from public, commercial, or not-for-profit funding agencies. The authors reported no potential conflict of interest. Data and codes are available under request and agreements.

References

Abdulhameed, O., Al-Ahmari, A., Ameen, W., Mian, S.H., 2019. Additive manufacturing: Challenges, trends, and applications. Adv. Mech. Eng. 11. https://doi.org/10.1177/1687814018822880

Bacciaglia, A., Ceruti, A., Liverani, A., 2022. Towards Large Parts Manufacturing in Additive Technologies for Aerospace and Automotive applications. Procedia Comput. Sci. 200, 1113–1124. https://doi.org/10.1016/j.procs.2022.01.311

BCN3D, n.d. BCN3D's signature IDEX Technology: Doubling productivity while halving costs.
Butt, J., Onimowo, D.A., Gohrabian, M., Sharma, T., Shirvani, H., 2018. A desktop 3D printer with dual extruders to produce customised electronic circuitry. Front. Mech. Eng. 13, 528–534. https://doi.org/10.1007/s11465-018-0502-1
Chinthavali, M., 2016. 3D printing technology for automotive applications, in: 2016 International Symposium on 3D Power Electronics Integration and Manufacturing (3D-PEIM). Presented at the 2016 International Symposium on 3D Power Electronics Integration and Manufacturing (3D-PEIM), IEEE, Raleigh, NC, USA, pp. 1–13. https://doi.org/10.1109/3DPEIM.2016.7570535
Claes, D., Tuyls, K., 2018. Multi robot collision avoidance in a shared workspace. Auton. Robots 42, 1749–1770. https://doi.org/10.1007/s10514-018-9726-5
Di Angelo, L., Di Stefano, P., Guardiani, E., 2020. An advanced GCode analyser for predicting the build time for additive manufacturing components. ACTA IMEKO 9, 30. https://doi.org/10.21014/acta_imeko.v9i4.728
DIN EN ISO/ASTM 52900:2021 Additive manufacturing — General principles — Fundamentals and vocabulary, n.d. . Beuth Verlag GmbH. https://doi.org/10.31030/3290011
Duran, C., Subbian, V., Giovanetti, M.T., Simkins, J.R., Beyette Jr, F.R., 2015. Experimental desktop 3D printing using dual extrusion and water-soluble polyvinyl alcohol. Rapid Prototyp. J. 21, 528–534. https://doi.org/10.1108/RPJ-09-2014-0117
Espalin, D., Alberto Ramirez, J., Medina, F., Wicker, R., 2014. Multi-material, multi-technology FDM: exploring build process variations: Rapid Prototyp. J. 20, 236–244. https://doi.org/10.1108/RPJ-12-2012-0112
Faria, C., Fonseca, J., Bicho, E., 2020. FIBR3DEmul—an open-access simulation solution for 3D printing processes of FDM machines with 3+ actuated axes. Int. J. Adv. Manuf. Technol. 106, 3609–3623. https://doi.org/10.1007/s00170-019-04713-y
Ferretti, P., Leon-Cardenas, C., Santi, G.M., Sali, M., Ciotti, E., Frizziero, L., Donnici, G., Liverani, A., 2021. Relationship between FDM 3D Printing Parameters Study: Parameter Optimisation for Lower Defects. Polymers 13, 2190. https://doi.org/10.3390/polym13132190
Ferro, C., Grassi, R., Seclì, C., Maggiore, P., 2016. Additive Manufacturing Offers New Opportunities in UAV Research. Procedia CIRP 41, 1004–1010. https://doi.org/10.1016/j.procir.2015.12.104
Fontaine, N., 2016. Modular user-configurable multi-part 3D layering system and hot end assembly. US20160067920A1.
Frutuoso, N., 2017. Toolpath Generation for a Multiple Independent Print Head System for Fused Deposition Modeling.
General Electric Company, 2013. GE jet engine bracket challenge [WWW Document]. URL https://grabcad.com/challenges/ge-jet-engine-bracket-challenge (accessed 4.23.20).
Guessasma, S., Zhang, W., Zhu, J., Belhabib, S., Nouri, H., 2015. Challenges of additive manufacturing technologies from an optimisation perspective. Int. J. Simul. Multidiscip. Des. Optim. 6, A9. https://doi.org/10.1051/smdo/2016001
Hiller, J.D., Lipson, H., 2009. Stl 2.0: A proposal for a universal multi-material additive manufacturing file format, in: In Mechanical and Aerospace Engineering. pp. 266–278.
Krishnanand, Soni, S., Nayak, A., Taufik, M., 2021. Generation of Tool Path in Fused Filament Fabrication, in: Agrawal, R., Jain, J.K., Yadav, V.S., Manupati, V.K., Varela, L. (Eds.), Recent Advances in Smart Manufacturing and Materials, Lecture Notes in Mechanical Engineering. Springer Singapore, Singapore, pp. 153–161. https://doi.org/10.1007/978-981-16-3033-0_14
Leite, M., Frutuoso, N., Soares, B., Ventura, R., 2018. Multiple Collaborative Printing Heads in FDM: The Issues in Process Planning. https://doi.org/10.26153/TSW/17207
Leite, Marco, Ventura, R., Boto, J., Frutuoso, N., Reis, L., Ribeiro, A.R., Soares, B., 2018. 3D printing of large parts using multiple collaborative deposition heads – a case study with FDM. Presented at the 3rd International Conference on Progess in Additive Manufacturing (Pro-AM 2018). https://doi.org/10.25341/D4WS3X
McPherson, J., Zhou, W., 2018. A chunk-based slicer for cooperative 3D printing. Rapid Prototyp. J. 24, 1436–1446. https://doi.org/10.1108/RPJ-07-2017-0150
Murr, L.E., 2016. Frontiers of 3D Printing/Additive Manufacturing: from Human Organs to Aircraft Fabrication†. J. Mater. Sci. Technol. 32, 987–995. https://doi.org/10.1016/j.jmst.2016.08.011
Pecho, P., Ažaltovič, V., Kandera, B., Bugaj, M., 2019. Introduction study of design and layout of UAVs 3D printed wings in relation to optimal lightweight and load distribution. Transp. Res. Procedia 40, 861–868. https://doi.org/10.1016/j.trpro.2019.07.121
Poudel, L., Blair, C., McPherson, J., Sha, Z., Zhou, W., 2020. A Heuristic Scaling Strategy for Multi-Robot Cooperative Three-Dimensional Printing. J. Comput. Inf. Sci. Eng. 20, 041002. https://doi.org/10.1115/1.4045143

Rais, M.H., Li, Y., Ahmed, I., 2021. Spatiotemporal G-code modeling for secure FDM-based 3D printing, in: Proceedings of the ACM/IEEE 12th International Conference on Cyber-Physical Systems. Presented at the ICCPS '21: ACM/IEEE 12th International Conference on Cyber-Physical Systems, ACM, Nashville Tennessee, pp. 177–186. https://doi.org/10.1145/3450267.3450545
Robotics, T., n.d. Titan Robotics' Cronus - Large-scale 3D printer.
Sigmund, O., Maute, K., 2013. Topology optimisation approaches: A comparative review. Struct. Multidiscip. Optim. 48, 1031–1055. https://doi.org/10.1007/s00158-013-0978-6
Ultimaker Cura, n.d.
Wachsmuth, J., 2008. Multiple Independent Extrusion Heads for Fused Deposition Modeling. Virginia Polytechnic Insitute and State University.
Wang, Y., Gu, Z., Song, L., Li, T., Cui, H., Lau, F.C.M., 2017. Speeding up 3D Printing Using Multi-Head Slicing Algorithms, in: 2017 5th International Conference on Enterprise Systems (ES). Presented at the 2017 5th International Conference on Enterprise Systems (ES), IEEE, Beijing, pp. 99–106. https://doi.org/10.1109/ES.2017.23
Zhang, X., Li, M., Lim, J.H., Weng, Y., Tay, Y.W.D., Pham, H., Pham, Q.-C., 2018. Large-scale 3D printing by a team of mobile robots. Autom. Constr. 95, 98–106. https://doi.org/10.1016/j.autcon.2018.08.004