



End-to-end path planning for homogeneous temperature fields in additive manufacturing

Iason Sideris ^{a,b,*}, Stephen Duncan ^d, Maicol Fabbri ^{a,c}, Francesco Crivelli ^b,
Mohamadreza Afrasiabi ^{a,c}, Markus Bambach ^{a,c}

^a Advanced manufacturing lab, D-MAVT, ETH Zurich, Sälimstrasse 101, Zurich, 8092, Switzerland

^b Group for Predictive Analytics, CSEM S.A., Untere Gründlistrasse 1, Alpnach, 6055, Obwalden, Switzerland

^c inspire AG, Technoparkstrasse 1, Zurich, 8005, Switzerland

^d Department of Engineering Science, University of Oxford, Parks Road, Oxford, OX1 3PJ, United Kingdom

ARTICLE INFO

Keywords:

Additive manufacturing

Path planning

Finite volume method (FVM)

Wire arc additive manufacturing (WAAM)

Reduced order modeling (ROM)

Monte Carlo Tree Search (MCTS)

ABSTRACT

This study explores a novel approach to path planning in deposition-based additive manufacturing, integrating the frequently overlooked process-induced temperature fields. Currently, existing approaches either ignore temperature effects entirely or only consider them in small-scale problems due to the high computational cost involved in predicting them and the combinatorial nature of path planning optimization. To address these challenges, the present work proposes an optimization pipeline that involves deriving a reduced order model from a finite volume method model with balanced truncation, using an analytical function to model the heat input and, calculating the steady-state response of the system to an arbitrary path using the Laplace transformation. Then, the optimization is transformed into a sequential decision-making problem and approximated with Monte Carlo tree search. The pipeline is validated through computational and experimental results, demonstrating its efficiency in managing large and complex geometries, as well as its resilience in overcoming the challenges posed by the simulation to reality gap.

1. Introduction

Additive manufacturing (AM) has emerged as a promising technique for producing complex and customized parts in various industries, including aerospace, automotive, and medical. Deposition-based AM processes, such as fused filament fabrication (FFF) for plastics and direct energy deposition (DED) for metals, have gained particular attention due to their high productivity, versatility and capability of processing a wide range of materials with low-cost equipment. In these techniques, the material is selectively deposited layer-by-layer through a nozzle, in the form of either a wire or a powder, and molten from an energy source, such as a heater in the case of FFF and an arc or a laser for DED.

Despite the numerous advantages of deposition-based AM processes, specific challenges persist, such as geometrical inaccuracies, residual stresses, distortions, and inhomogeneous material properties. These complications frequently stem from non-uniform temperature distributions and inconsistent material deposition, largely due to ineffective toolpath strategies. Therefore, the goal of the present study is to devise a robust and efficient path planning algorithm that overcomes

these challenges and guarantees the high quality, accuracy, and low build-time of the final parts.

Path planning in deposition-based AM involves determining a sequence of tool movements and deposition parameters to accurately replicate target geometries, minimize production time, and meet the specified part requirements. Effective path planning is crucial for manufacturing high-quality parts with minimal defects and uniform properties, and it must account for factors such as heat input, deposition rates, and cooling rates. However, simultaneously incorporating all these factors presents a significant challenge. This complexity arises because estimating the impacts of potential paths can be computationally intensive, and the number of viable deposition strategies increases exponentially with the complexity of the parts being produced. Consequently, current literature often addresses different aspects of path planning separately in order to simplify the problem and mitigate this computational bottleneck.

Geometric accuracy is essential in AM path planning, with much of the literature addressing this issue. A common approach is to plan and control the kinematics of the deposition head, leading to simple and fast algorithms that overlook dynamic phenomena and potential defects.

* Correspondence to: Advanced Manufacturing Laboratory, ETH Zurich, 8005 Zurich, Switzerland.

E-mail address: isideris@ethz.ch (I. Sideris).

<https://doi.org/10.1016/j.jmatprotec.2024.118364>

Received 19 October 2023; Received in revised form 28 February 2024; Accepted 3 March 2024

Available online 7 March 2024

0924-0136/© 2024 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Nomenclature

Symbol

θ	Temperature
θ_a	Ambient temperature
ρ	Density
C_p	Heat capacity
K	Heat conductivity
Q	Heat forcing term
q_{hs}	Heat from heat-source
q_{bw}	Heat loss to solid boundaries
q_{bc}	Heat loss to environment
Z_r	Reducing transformation
v	Reduced state
A, B, B_d, C, D	Time-dependent dynamic matrices
A_*, B_*, B_{b*}, C_*	Fixed dynamic matrices
A_r, B_r, B_{rb}, C_r	Reduced dynamic matrices
\tilde{q}_{hs}	Volume-wise input model
A_{in}, Δ	Magnitude and phase of input model
E_{in}	Thermal response
\mathcal{V}	MCTS leaf value
S	MCTS leaf state
I	Path plan
(f, s, d)	Tuple of feature, welding start point and welding direction

Schmitz et al. (2021) introduce a path planning approach for Wire-arc additive manufacturing (WAAM) directly considering robot kinematics to ensure all scheduled tool poses are reachable. The planned paths are smoothed to respect joint velocity limits, though not optimized for deposition defects or process dynamics. Ščetinec et al. (2021) tackle variable layer heights in WAAM with an online control scheme and tool path replanning for large shell geometries. They use an arc-current controller and a re-slicing algorithm to correct deviations from the original CAD models and demonstrated their performance in a 160 mm high part case study. Furthermore, Bui et al. (2019) develop a tool path optimizer for collaborating print-heads using Tabu Search, focusing on collision-free, efficient deposition. Their algorithm shows improved production time in simulations, maintaining geometric accuracy, yet it mainly considers process kinematics, neglecting dynamic aspects.

To combine high productivity and reduce the number of transient welding states, i.e. deposition starts and stops, which are associated with defects, a set of algorithms has arisen targeting to devise a continuous path for every layer. However, these algorithms are limited by their uneven temperature distributions and the geometries they can be applied because they require that the layer geometry is turned into an Eulerian graph. Nguyen et al. (2020) formulate a strategy to achieve continuous WAAM tool paths for rib-web structures by converting the layers into Eulerian graphs, which not only accelerated deposition but also helped in reducing welding defects. Building on this study, Yamamoto et al. (2022) propose a path planning algorithm for fused deposition modeling with continuous carbon fiber-reinforced filaments, generating single-stroke paths for each layer using Hierholzer's Algorithm. Additionally, Xia et al. (2023) introduce a lightweight design algorithm, converting dense parts into beam structures and optimizing for a globally continuous path, thereby improving print quality while reducing underfilled and overfilled areas. All these studies, which consider manufacturing practicalities, have been experimentally validated, demonstrating improvements over traditional methods.

Continuous path methods offer high productivity and fewer defects but still face challenges like limited applicability to complex geometries, uneven temperature distribution, and difficulty in creating paths

for non-2.5D shapes. Feature-based path planning methods offer a potential solution to the applicability problems of continuous welding methods, while maintaining high productivity levels. However, they often overlook the thermal aspect of the process as this requires solving a discrete optimization problem whose size grows with the complexity of the part. In this realm, Diourté et al. (2021) introduce a feature-based, continuous, 3D strategy for WAAM, where layers can vary in thickness and shape with an adaptive wire feed rate, enabling the fabrication of intricate geometries. This approach allows for complex shapes with high precision and fewer welding starts and stops, surpassing the capabilities of 2.5D strategies. Expanding upon this concept, Petrik and Bambach (2023) develop a reinforcement learning algorithm for thin-walled WAAM structures, using proximal policy optimization to define the welder's moves and process parameters like welding speed and wire feed rate, reducing user input and promoting autonomy. Similarly, Michel et al. (2019) create a feature-based path planner to minimize deposition artifacts by segmenting layers into distinct features with unique paths, though their work does not address the sequencing of the welder's movements, a key factor in temperature distribution and stress management.

The significance of path planning in influencing temperature distributions and mechanical performance has been extensively studied, yet a complete solution remains elusive. Initial research concentrated on finding deposition patterns that minimize stresses in specific geometries but fell short in generalizing. Foroozmehr and Kovacevic (2010) and Venturini et al. (2016) explore different deposition patterns, focusing on the balance between stress distributions and productivity. Although they evaluated various candidate paths through computational or experimental means, their findings are primarily applicable to geometries similar to those studied.

Recently, there has been a surge in the development of deposition strategies that are adaptable to a wide range of geometries, with a particular focus on incorporating thermomechanical effects. Despite these advancements, such strategies frequently encounter scalability challenges. These challenges stem either from the computationally intensive nature of evaluating process-induced fields or from the overwhelming number of potential paths that must be assessed. Consequently, the application of these advanced strategies often remains confined to relatively simple examples. Xia et al. (2020) introduce a tool path planning strategy for FFF, wherein the tool paths are aligned with the maximum principal stresses, determined via the finite element method (FEM). This approach not only improves mechanical performance but also maintains low deposition time by minimizing nozzle jumps using Dijkstra's algorithm. Their tests on PLA-printed specimens showed the effectiveness of the strategy, but challenges remain in maintaining consistent deposition conditions, especially with high-performance materials, due to the need for stable preheat temperatures which are not preserved by the deposition policy. Additionally, Yang et al. (2023) present a sophisticated approach to mitigate temperature inhomogeneity in AM by exploring the inverse heat placement problem. Specifically, they try to find incremental adjustments to a predefined strategy, e.g. raster, and discover the optimal deposition parameters that even out the maximum temperatures estimated on a set of observation points. This technique can effectively level out temperatures in scanned areas. However, it introduces some variability in overlaps between subsequent passes, which might lead to issues like pores or material buildup and gets computationally demanding when adding more degrees of freedom and observation points. Also, Alhakeem et al. (2023) focus on homogenizing temperature distributions in WAAM. Their algorithm creates paths by depositing weld beads in the coolest substrate areas, evaluating the thermal footprint of each path using a simplified FEM model. While this heuristic offers better outcomes than arbitrary paths, its effectiveness is constrained by the exhaustive search required and does not ensure optimal path generation, as it chooses the bead order over a short horizon without considering long-term selection implications.

Finally, the seminal work of [Beisegel et al. \(2021\)](#) explores the use of mixed integer programming (MIP) to homogenize the temperature fields. In their article, MIP is employed to find the optimal sequence from a set of pre-defined moves, akin to those in [Michel et al. \(2019\)](#), yet they also encounter similar scalability issues like the rest of the methods. Their method involves approximating the heat equation solution to define the objective function and constraining path searches to ensure full surface coverage. While MIP can provide solutions with optimality certificates, unlike heuristics or algorithms like those by [Al-hakeem et al. \(2023\)](#), it demands numerous cost function evaluations. This necessity led the authors to simplify the problem domain, fix preheat temperatures for entire layers, and create cost functions that only implicitly promote temperature uniformity. Moreover, to keep the number of possible paths within computationally feasible limits, they held the number of predefined moves relatively low. For instance, optimizing a part with 10 pockets required limiting the moves to 10, yielding 3,628,800 possible paths and requiring about 190,800 s for computation. Such findings demonstrate that MIP could not scale to more complex geometries, where more than 10 moves are required to describe the whole part.

While the importance of the deposition strategy-induced temperature fields is recognized in the literature and their connection to stresses and material overflow is established, current path planning methods either dismiss the thermal aspect of planning or try to reduce it by simplifying the possible number of moves and by using heuristics. These limitations arise from two main factors:

- evaluating the thermal response of a system is computationally intensive, making iterative optimization algorithms impractical
- path planning NP-hard, which is a class of computational problems for which no efficient solution algorithm is known, and solving them becomes exceptionally time-consuming as the problem size increases. Consequently, introducing additional complexity rapidly increases the size of the problem. This makes the application of advanced exact optimization infeasible for practical applications, which forces the adoption of approximate methods

To address these issues, we introduce a novel approach that tackles the limitations present in the current state of the art. Our proposed solution consists of three main components:

- a pipeline that derives a fast reduced order model (ROM) from a finite volume method (FVM) model, which requires no experimental data to be obtained
- an analytical model of the heat input and the calculation of the steady-state response of the system using Laplace transformations
- turning the optimization to a sequential decision-making problem, modeling it as a state tree and then using the Monte Carlo Tree Search (MCTS) algorithm, which, unlike exact MIPs, avoids unnecessary evaluations of low-performing candidates, and quickly discovers near-optimal solutions, which allows approximate optimal paths to be determined more quickly

To illustrate the integration of these three components, we start with an FVM model that simulates the deposition of a part, where new elements are activated according to a heating schedule for each layer. Then, the dynamics of the system are fixed at an appropriate point in time, and the system is transformed to its reduced form, which only considers the most important aspects of its input–output behavior. Following this, the Laplace transform is utilized to efficiently estimate, via the ROM, the temperature fields that would result from repeatedly applying the same heating schedule in the future. Leveraging this approximation, MCTS is applied to find the heating plan that best achieves a specific goal, in this case, to minimize the standard deviation of the predicted temperature fields on the deposition layer. As a result, this optimization scheme seeks to discover the heating plan that deposits a layer that keeps the welding preheat temperature as constant as possible.

An alternative to ROM could be found in machine learning, where many fast models either trained on simulations ([Mozaffar et al., 2021](#)) or on experimental data ([Sideris et al., 2023](#)) exist. However, these methods have practical downsides. Models akin to [Mozaffar et al. \(2021\)](#) require extensive computation time for dataset creation and training, leading to significant time costs. Furthermore, their effectiveness is limited by the data's relevance and they do not necessarily improve the optimization performance, as indicated in Sections 5 and 6. Meanwhile, methods similar to [Sideris et al. \(2023\)](#), which involve creating a model from a part before optimizing its path, contradict the additive manufacturing principle of achieving the right result on the first try, thus impacting efficiency and waste reduction.

In the subsequent sections, this article is structured as follows. Section 3 provides a comprehensive overview of the proposed pipeline, beginning with the formulation of the optimization problem in Section 2. Next, the system modeling is described in Section 3.1, followed by the derivation of the ROM in Section 3.2 and the steady-state calculation in Section 3.3. The optimization algorithm is then presented in Section 3.4, accompanied by the corresponding results and the experimental setup used to obtain them, as detailed in Sections 4 and 5. Finally, a comprehensive discussion and conclusive remarks on this study can be found in Sections 6 and 7 respectively.

Additional material on the project can be found on <https://iss1995.github.io/path-planning-homogeneous-temperature-fields>.

2. Problem definition

The deposition of a layer can be decoupled in a series of feature-based deposition patterns, as proposed in [Michel et al. \(2019\)](#). However, this approach introduces a new layer of complexity as each feature can typically be deposited in multiple ways, and the order of feature deposition must be determined. The choice of feature path and order significantly influences the occurrence of defects, material overflow, and residual stresses. Consequently, an optimization problem is formulated to identify the optimal deposition pattern for each feature and the most appropriate feature order.

In the field of operational research, such combinatorial problems are called flexible job shop scheduling problems ([Xiong et al., 2022](#)), and they constitute a subcategory of machine scheduling problems ([Johnson, 1954](#)). In the special case where the deposition pattern for each feature is fixed, then the problem is transformed into a single machine scheduling problem ([Lawler et al., 1993](#)) which corresponds to the optimization problem addressed by [Beisegel et al. \(2021\)](#).

The first step to formulating the optimization problem is to select a norm that reflects the traits of the optimal path. In this work, we select to minimize the maximum spatial standard deviation of the process-induced temperature profile, occurring at the deposition layer

$$\begin{aligned} \min \quad & \|\text{std}(\theta_d(t))\|_\infty \\ \text{over} \quad & t_0 \leq t \leq t_0 + T \end{aligned} \quad (1)$$

where $t_0 = nT$ is the time point that the deposition of the n th layer starts, and T is the total time needed to print the layer, including the resting time between two layers. The term $\text{std}(\theta_d(t))$ is the standard deviation over x, y , and z at a time point t , of the temperature field developed on the deposition layer $\theta_d(t) \in \Omega \subset \mathcal{R}^3$

$$\text{std}(\theta_d(t)) = \sqrt{\frac{1}{V} \iiint_{\Omega} (\theta_d(x, y, z, t) - \mu(t))^2 dV}$$

with $\mu(t) = \frac{1}{V} \iiint_{\Omega} \theta_d(x, y, z, t) dV$ being the mean of the field, and V being the volume of Ω . The choice of the objective function is motivated by the strong link between the temperature profiles and residual stresses, microstructure, and material overflow. Achieving a homogeneous temperature field suggests that the welding conditions are constant all over the deposition plane and can contribute to the following:

1. mitigating the induced thermal stresses in the final part (Li et al., 2018)
2. achieving similar microstructural characteristics in the whole volume (Akram et al., 2018)
3. reducing material overflow and guaranteeing that each feature will have the intended shape (Alhakeem et al., 2023)

The infinity norm is chosen to strengthen the robustness of the solutions, although it is not the only norm that could deliver the desired results. Other compelling choices of norms could be the 2-norm of the standard deviation or even the infinity norm of the thermal gradients.

Also, in this study n is chosen sufficiently large, to exclude transient responses from θ_d since they heavily depend on the preheating of the deposition bed, which is not easy to control in metal deposition-based AM processes. For large n values the system is considered to have reached a steady-state, meaning that repeating a path plan for two subsequent layers results in the same standard deviation

$$\text{std}(\theta_d(t)) = \text{std}(\theta_d(t + T))$$

over $t_0 \leq t \leq t_0 + T$

As a next step, the path I is defined. A path plan $I = \{i_1, i_2, \dots, i_N\}$ comprises each feature's deposition order and selected path, which will be noted as a move i_n , and essentially describes the motion of the moving heat source. More specifically, i_n is the n th move in the path and depicts a certain variant of a feature path. The same feature could have been deposited with the same order but with a different feature path i'_n , or with the same feature path but on a different order $i_{n'}$. An example of features and possible feature paths can be found in Figs. 5(a) and 5(b).

Each move is associated with its starting time t_{i_n} , which can be uniquely defined as

$$t_{i_n} = \sum_{i=1}^{i_n-1} t_i \tag{2}$$

with $T = \sum_{i=1}^{i_N} t_i$ and t_i being the length of each interval. The timings and the choice of moves i_n uniquely define the heat input from the moving source, $q_{hs}(x, y, z, t; I)$, which then induces the temperature profiles θ_d .

However, the selection of a path I must satisfy certain constraints to ensure a successful build. Specifically, the chosen moves i_n must result in a layer fully covered with material, without any overlapping segments. In other words, I must transverse the entire deposition area exactly once. All the paths satisfying these constraints are called *permissible*, and the set of all such paths for a layer is denoted as J . With J established, the optimization problem of finding the optimal path plan can be formulated as follows:

$$I^* = \underset{I \in J}{\text{argmin}} \left\| \text{std}(\theta_d(q_{hs}(t; I))) \right\|_{\infty} \tag{3}$$

over $t_0 \leq t \leq t_0 + T$

Thus the objective is to identify the optimal sequence and moves that yield a heating plan, minimizing the value of the spatial standard deviation profile. The complexity of this problem is significant as the cardinality of J increases exponentially with added moves, which presents a substantial computational challenge for parts with intricate geometries commonly encountered in industrial settings.

3. Methods

3.1. Finite volume method

FVM models are commonly employed in the AM field to represent the intricate dynamics involved in the process due to their low computational cost compared to their FEM counterparts. The FVM method transforms a nonlinear partial differential equation (PDE) into a linear system of ordinary differential equations (ODEs) by discretizing

the problem domain into a finite number of control volumes and applying conservation principles to each volume. This results in a set of linear algebraic equations called the weak form, that approximates the original nonlinear PDE. In this work, we follow the FVM thermal model of Fabbri et al. (2024), outlined in the following subsection.

The goal is to derive an FVM model from the heat transfer equation for an isotropic and homogeneous medium forced by an external heat source

$$\rho c_p(\theta) \frac{\partial \theta}{\partial t} = \nabla \cdot (K(\theta) \nabla \theta) + Q \tag{4}$$

where ρ is the density, θ is the temperature, t is time, $c_p(\theta)$ is the temperature-dependent heat capacity, $K(\theta)$ is the temperature-dependent heat conductivity, and Q is the forcing heat source term. In this paper, we focus on part-scale temperature distributions and disregard melt pool phenomena, as they primarily contribute to the fields of the melt pool area.

For AM processes, where the heat source is not stationary in space, the term Q can be approximated as

$$Q = q_{hs} + q_{bw} + q_{bc} \tag{5}$$

where q_{hs} represents the input heat from the moving heat source following the Goldak model (Goldak et al., 1984), q_{bw} is the heat loss to the surrounding walls through conduction and q_{bc} is the heat loss to the environment through radiation and convection:

$$q_{hs} = \begin{cases} \frac{P f_f 6 \sqrt{3}}{a_f b c \pi \sqrt{\pi}} \exp\left(-3 \left(\frac{\xi'^2}{a_f^2} + \frac{\eta'^2}{b^2} + \frac{\chi'^2}{c^2}\right)\right), & \xi' \geq 0 \\ \frac{P f_r 6 \sqrt{3}}{a_r b c \pi \sqrt{\pi}} \exp\left(-3 \left(\frac{\xi'^2}{a_r^2} + \frac{\eta'^2}{b^2} + \frac{\chi'^2}{c^2}\right)\right), & \xi' < 0 \end{cases} \tag{6}$$

In this equation, P is the heat input power, a , b , c_f , c_r , f_f , f_r are parameters defining the shape of the volume and χ' , ξ' , η' is the local coordinate system for the moving heat-source.

q_{bw} : The term q_{bw} represents the energy transfer between the part and solid boundary walls, such as the deposition table. For an elementary surface ds , the term can be expressed as

$$q_{bw} = h_s^c (\theta_s^w - \theta_s) ds$$

where h_s^c is the thermal conductivity between the two objects at ds , θ_s and θ_s^w are the part and wall temperature profiles.

q_{bc} : The term q_{bc} represents the energy transfer between the part and the ambient environment through convection and radiation. For an elementary surface ds , the term can be expressed as

$$q_{bc} = \hat{h}_s (\theta_s, \theta_a) (\theta_a - \theta_s) ds$$

$$\hat{h}_s = \sigma \epsilon (\theta_a + \theta_s) (\theta_a^2 + \theta_s^2) + h$$

with \hat{h}_s being an equivalent heat convection parameter, including both convection and radiation terms (Sideris et al., 2023), σ the Stefan-Boltzmann constant, ϵ the emissivity, h being the convection coefficient, and θ_a being the ambient temperature.

Next, the domain is discretized into a finite number of rectangular hexahedra control volumes V_i . By applying the conservation principle over each volume and employing the divergence theorem, Eq. (4) is transformed to

$$\int_{V_i} \rho c_p(\theta) \frac{\partial \theta}{\partial t} dv = \int_{\partial V_i} -K(\theta) \nabla \theta \cdot \mathbf{n} ds + \int_{V_i} Q dv \tag{7}$$

Here \mathbf{n} is the unit normal vector of the surface ∂V_i . The usage of rectangular hexahedra allows approximating the surface integral as

$$\int_{\partial V_i} K(\theta) \nabla \theta \cdot \mathbf{n} ds \approx \sum_{j \in \mathcal{N}(i)} K_{ij} \frac{\theta_j - \theta_i}{d_{ij}^2} A_{ij}$$

where K_{ij} is the thermal conductivity at the midpoint between the centroids of control volumes V_i and V_j , d_{ij} is the distance between the

centroids, A_{ij} is the interface area between the control volumes, and $\mathcal{N}(i)$ represents the set of neighboring control volumes of the volume V_i . Substituting back to Eq. (7) and by calculating the integrals, the thermal evolution of each control volume V_i is given by

$$\rho c_{p,i} \frac{\partial \theta_i}{\partial t} V_i \approx \sum_{j \in \mathcal{N}(i)} K_{ij} \frac{\theta_j - \theta_i}{d_{ij}^2} A_{ij} + Q_i V_i \quad (8)$$

Here, θ_i is the temperature at the centroid of the control volume V_i , S_{ij} is the interface area between control volumes V_i and V_j , d_{ij} is the distance between the centroids of control volumes V_i and V_j , and $Q_i = q_{hs,i} + q_{bw,i} + q_{bc,i}$ is the average value of the forcing term in volume V_i .

Redistributing the terms in Eq. (8), and gathering all the temperatures at the volume centroids $i \in [1, 2, \dots, N]$, a system of discretized ODE equations can be derived

$$\dot{\theta}(t) = \mathbf{A}(\theta(t), t)\theta(t) + \mathbf{B}(t)q_{hs}(t) + \mathbf{B}_a(\theta(t), t)\theta_a(t) + \mathbf{B}_w(t)\theta_w(t) \quad (9)$$

where $\theta(t) \in \mathcal{R}^N$ is the vector containing the temperatures at the centroids of all the control volumes at some point in time, $\mathbf{A}(\theta(t), t) \in \mathcal{R}^{N \times N}$ is the system dynamics' matrix, $\mathbf{B}(t) \in \mathcal{R}^{N \times U}$ is the matrix mapping the input heat from the moving source to the volumes, $\mathbf{B}_a(\theta(t), t) \in \mathcal{R}^N$ accounts for ambient environment boundary conditions, $\mathbf{B}_w(t) \in \mathcal{R}^N$ addresses solid surfaces boundary conditions, and $q_{hs}(t) \in \mathcal{R}^U$ is the heat input vector. In this work, we consider that $\theta_w(t)$ and $\theta_a(t)$ are equal and constant, transforming Eq. (9) to

$$\dot{\theta}(t) = \mathbf{A}(\theta(t), t)\theta(t) + \mathbf{B}(t)q_{hs}(t) + \mathbf{B}_b(\theta(t), t)\theta_a \quad (10)$$

with $\mathbf{B}_b(\theta(t), t) = \mathbf{B}_a(\theta(t), t) + \mathbf{B}_w(t)$ introducing the boundary conditions, and θ_a being the constant ambient temperature.

The resulting system of ODEs in Eq. (10) can be considered as a state-space model whose states are the degrees of freedom of the FVM model

$$\begin{aligned} \dot{\theta}(t) &= \mathbf{A}(\theta(t), t)\theta(t) + \mathbf{B}(t)q_{hs}(t) + \mathbf{B}_b(\theta(t), t)\theta_a \\ \mathbf{y}(t) &= \mathbf{C}(t)\theta(t) + \mathbf{D}(t)q_{hs}(t) \end{aligned} \quad (11)$$

where the states and the output $\mathbf{y}(t) \in \mathcal{R}^N$ corresponds to the temperatures at the centroid of the control volumes, leading to $\mathbf{C}(t) = \mathbf{I}_N$ and $\mathbf{D}(t) = \mathbf{0}$. The system in Eq. (11) is non-linear and time-variant as it grows over time with newly deposited layers, and the boundary conditions depend on $\theta(t)$. To extract the ROM in Section 3.2 the system should be transformed into a linear time-invariant (LTI) state-space model and for therefore two simplifications are made:

1. we consider a moment t^* in time, where the thermal response of the system does not change significantly with the addition of more material. The time-step t^* is approximated numerically by monitoring the dominant singular values of the matrix $A(t^*)$ and will be referenced as the *transition time* of the geometry.
2. the transition time t^* is selected between 2 subsequent layers, where the layer geometry is fully deposited, all material is solid and temperatures have settled.

In practice, to determine t^* , the singular values of the system are computed at the final timestep of the cooling time following the deposition of a layer, where $d\theta/dt \approx \text{const}$, and the radiation effect is trivial compared to convection. Afterwards, the M most significant modes are compared. Finally, t^* is selected as the timestep where there is no notable change between the singular values corresponding to the M modes of 2 subsequent layers. The system dynamics at t^* are denoted as $\mathbf{A}_* = \mathbf{A}(\theta(t^*), t^*)$, $\mathbf{B}_* = \mathbf{B}(t^*)$, $\mathbf{B}_{b*} = \mathbf{B}_b(\theta(t^*), t^*)$ and, $\mathbf{C}_* = \mathbf{C}(t^*)$, and essentially dictate the reheating of the system

$$\begin{aligned} \dot{\theta}(t) &= \mathbf{A}_*\theta(t) + \mathbf{B}_*q_{hs}(t) + \mathbf{B}_{b*}\theta_a \\ \mathbf{y}(t) &= \mathbf{C}_*\theta(t) \end{aligned} \quad (12)$$

3.2. Model reduction

FVM has several benefits, including its computational efficiency and ability to handle non-linear systems. However, these models inherently possess many degrees of freedom, which stem from the spatial discretization selected for the analysis. The finer the discretization, the more volumes are introduced into the system, leading to an increased number of degrees of freedom.

The state-space models of Eqs. (11) and (12) are considered to have full-state observability, meaning that the temperature at every finite volume center is known. This results in a system with as many outputs as elements in the model. Additionally, the moving heat source activates multiple degrees of freedom leading to a high dimensional input. Consequently, this leads to a multiple-input, multiple-output (MIMO) system of very high order. The sheer size and complexity of such MIMO systems pose significant challenges in terms of computational efficiency and do not fit with iterative optimization frameworks.

Given the difficulty of optimizing these large-scale systems, it becomes essential to investigate the reduction of FVM models to a lower order. The objective is to find a ROM that strikes a balance between model accuracy and computational efficiency, enabling the numerical optimization of the process plan.

One tool that enables this reduction is balanced truncation (BT) (Brunton and Kutz, 2019; Lall et al., 2002; Moore, 1981). BT combines proper orthogonal decomposition with a balancing transformation to obtain a ROM that captures the r most observable and controllable modes of the original system. The balancing part of BT refers to a linear transformation applied to an LTI system that diagonalizes its controllability and observability Grammians simultaneously. Balancing seeks to identify the states that are most important for the input-output behavior of the system, allowing for a more effective model reduction while preserving the system's dynamic characteristics.

However, computing the BT of an $N \times N$ ODE system, with $\mathbf{C} = \mathbf{I}_N \in \mathcal{R}^{N \times N}$ and $\mathbf{B} \in \mathcal{R}^{N \times U}$ where both N and U are large, is a computationally and memory intensive task, with complexities of $O(N^3)$ and $O(N^2)$ respectively. Scaling this method to large FVM models is cumbersome and requires attention. A remedy for this is to introduce another step in the reduction procedure, where first the major modes in the matrix \mathbf{A}_* are identified, and then, the balanced reduction is only performed on them. For a stable FVM model, the matrix \mathbf{A}_* is negative definite leading to negative and real eigenvalues. As a result, the modes associated with high magnitude eigenvalues dissipate fast, while slow modes with long settling times represent the dominant dynamics, which are responsible for the most significant behavior of the system. Thus, the sub-part of the system composed by its M slowest eigenmodes includes the M most important modes for the process.

Nevertheless, defining the value of M is not easy. For this reason, M is selected to be large, in this case containing eigenvalues ranging across three orders of magnitude, and then BT is employed to transform this sub-system to a more condensed representation, i.e. an r -degree ROM. Finally, to facilitate the rest of the analysis, the ROM is further diagonalized resulting in a decoupled system of r ODEs. If $\mathbf{Z}_{11} \in \mathcal{R}^{N \times M}$ is the matrix whose columns are the M slowest eigenmodes of \mathbf{A}_* , $\Phi^* \in \mathcal{R}^{r \times M}$ includes the r first rows of the inverse balancing transformation and $\mathbf{Z}_3 \in \mathcal{R}^{r \times r}$ diagonalizes the reduced system, then the total reducing transformation from θ to v can then be expressed as

$$\begin{aligned} v(t) &= \mathbf{Z}_r\theta(t) \\ \mathbf{Z}_r &= \mathbf{Z}_3^{-1}\Phi^*\mathbf{Z}_{11}^T \end{aligned} \quad (13)$$

and Eq. (12) is transformed to

$$\begin{aligned} \dot{v}(t) &= \mathbf{A}_r v(t) + \mathbf{B}_r q_{hs}(t) + \mathbf{B}_{rb}\theta_a \\ \mathbf{y}(t) &= \mathbf{C}_r v(t) \end{aligned} \quad (14)$$

where $\mathbf{A}_r \in \mathcal{R}^{r \times r}$ is the diagonal matrix including the ROM eigenvalues, $\mathbf{C}_r = \mathbf{Z}_r^T \in \mathcal{R}^{N \times r}$ is the measurement matrix, $\mathbf{B}_r = \mathbf{Z}_r \mathbf{B}_* \in \mathcal{R}^{r \times U}$

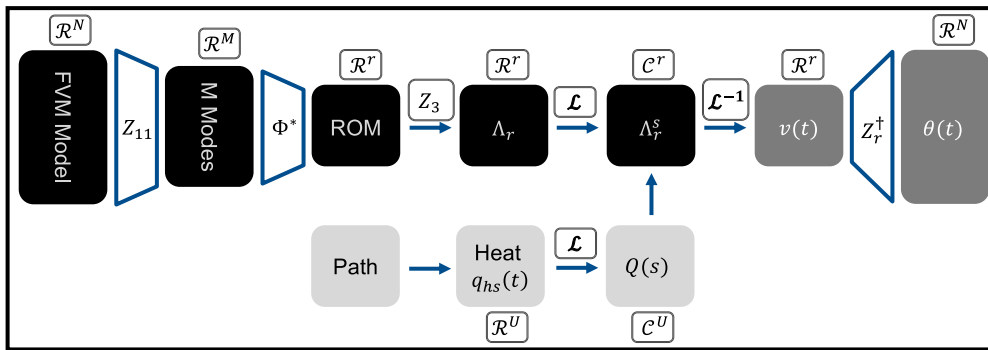


Fig. 1. Model reduction and steady-state response pipeline. The original finite volume method (FVM) model, characterized by N degrees of freedom, undergoes reduction to r dimensions, and subsequent projection into the Laplace space. Then, the steady-state response of the reduced order model (ROM) can be computed for any given trajectory, with the solution then projected back into the original space. The superscript \dagger denotes the pseudo-inverse of the transformation matrix.

and $\mathbf{B}_{r,b} = \mathbf{Z}_r \mathbf{B}_{b*} \in \mathcal{R}^r$ are the matrices mapping the input heat and boundary conditions to the reduced state.

The exact derivation of each transformation and a thorough description of every reduction step can be found in the Appendix A.

3.3. Heat input model

To improve computational efficiency in path evaluation, a low-dimensional approximation of the FVM model introduced in Eq. (10) was formulated in Eq. (14). Nonetheless, while the ROM of Eq. (14) reduces the number of degrees of freedom in the system, it does not alleviate the need for costly time integration. To further enhance computational performance, the heat input is analytically modeled and the steady-state response of Eq. (14) is determined via the Laplace transformation (Ogata, 2010). Determining the temperature evolution via the steady-state response is significantly faster than integrating over time as it requires significantly fewer operations to obtain.

The ROM is actuated by the heat source $q_{hs}(t)$, as outlined in Eq. (6), which describes the thermal footprint of the tool based on its position. However, calculating the Laplace transformation requires the definition of the heat source contribution $\tilde{q}_{hs}(t)$ to each control volume V_i . Each element of $\tilde{q}_{hs}(t)$, namely \tilde{q}_i , signifies the heat input at V_i . Considering that in a stable FVM model, the velocity of the tool is very high compared to the dimensions of V_i , $\tilde{q}_{hs}(t)$ is approximated as a series of impulses. During each move i_n of any permissible path plan $I = \{i_0, i_1, \dots, i_{i_n}\} \in \mathcal{J}$, each V_i can be activated once by the heat source at a specific timestep t_{i,i_n} . The heat input at a control volume can be mathematically expressed as

$$\tilde{q}_i = \sum_{i_n \in I} a_{i,i_n} \delta(\tau_{i,i_n}) \quad (15)$$

In this expression, δ is the Dirac impulse function, $\tau_{i,i_n} = t - t_{i,i_n} - nT$ is the phase of the impulse, T is the layer deposition time interval and n is the number of already deposited layers. For each V_i , the terms t_{i,i_n} and a_{i,i_n} , correspond to the activation time and the amplitude of the heat impulse, during the move i_n . To guarantee the energy conservation of the original heat input, the amplitude is mathematically given as the integral of the input heat over a small time interval around τ_{i,i_n} :

$$a_{i,i_n} = \int_{\tau_i - \epsilon}^{\tau_i + \epsilon} q_{hs,i} d\tau$$

Aggregating the heat inputs \tilde{q}_i for all control volumes yields the total heat input \tilde{q}_{hs} . This can be expressed as:

$$\tilde{q}_{hs} = \sum_{i_n} \mathbf{A}_{i_n} \Delta(\tau) \quad (16)$$

with $\mathbf{A}_{i_n} = \text{diag}(a_{0,i_n}, a_{1,i_n}, \dots, a_{U,i_n}) \in \mathcal{R}^{U \times U}$, $\tau \in \mathcal{R}^U$ being the stacked magnitude and phases, and $\Delta(\tau) \in \mathcal{R}^U$ representing the stacked Dirac

functions, centered around the respective activation timestep, for every actuated node

$$\Delta(\tau) = \begin{bmatrix} \delta(\tau_{1,i_n}) \\ \delta(\tau_{2,i_n}) \\ \vdots \\ \delta(\tau_{U,i_n}) \end{bmatrix} = \begin{bmatrix} \delta(t - t_{1,i_n} - nT) \\ \delta(t - t_{2,i_n} - nT) \\ \vdots \\ \delta(t - t_{U,i_n} - nT) \end{bmatrix}$$

This mathematical model illustrates the actuation mechanism of the moving heat source across the control volumes. As the heat source moves, it imparts a specific amount of energy to each control volume it crosses, leading to a transient heat transfer process. To analyze the impact of the heat source on the system, Eq. (16) is plugged back into Eq. (14)

$$\dot{v}(t) = \Lambda_r v(t) + \mathbf{B}_r \sum_{i_n} \mathbf{A}_{i_n} \Delta(\tau) + \mathbf{B}_{r,b} \theta_a \quad (17)$$

Given that the path plan I is periodically applied to the part at intervals of T , the long-term or steady-state response of Eq. (17) to I is also periodic, meaning that $v(t) = v(t+T)$. The steady-state can be calculated via the Laplace transformation as

$$v(t) = \sum_{i_n} \mathbf{E}_{i_n}(\tau) + \mathbf{E}_0(\tau_0) - \Lambda_r^{-1} \mathbf{B}_{r,b} \theta_a \quad (18)$$

where $\mathbf{E}_{i_n,m}(\tau) = \mathbf{B}_{r,m} \mathbf{A}_{i_n} (\mathbf{H}(\tau) \circ e^{\lambda_m \tau}) \in \mathcal{R}$ gives the response of each mode $m = 1 \dots r$, $\mathbf{E}_{0,m}(\tau_0) = v_m(nT) \exp(\lambda_m \tau_0) \in \mathcal{R}$ gives the contribution of the initial conditions at the beginning of the period $n+1$, $\Lambda_r^{-1} \mathbf{B}_{r,b} \theta_a$ is constant and corresponds to an offset for the boundary conditions, \circ denotes the Hadamard product, the index m indicates the m th row of the corresponding matrix and, and $\tau_0 = t - nT$ sets the time spent on the current layer. The vectors $\mathbf{H}(\tau) \in \mathcal{R}^U$ and $e^{\lambda_m \tau} \in \mathcal{R}^U$ fabricate the impulse response of each mode, and are given by

$$\mathbf{H}(\tau) = \begin{bmatrix} h(\tau_{1,i_n}) \\ h(\tau_{2,i_n}) \\ \vdots \\ h(\tau_{U,i_n}) \end{bmatrix}, e^{\lambda_m \tau} = \begin{bmatrix} \exp(\lambda_m \tau_{1,i_n}) \\ \exp(\lambda_m \tau_{2,i_n}) \\ \vdots \\ \exp(\lambda_m \tau_{U,i_n}) \end{bmatrix}$$

with $h(\tau_{i,i_n})$ being the Heaviside function activated at τ_{i,i_n} . For a detailed derivation of the steady-state response, the interested reader may refer to Appendix B. Also, an outline of the reduction steps and the steady-state response calculation is illustrated in Fig. 1.

Eq. (18) enables the direct calculation of the steady-state response of the state-space system in Eq. (11) to an arbitrary path, skipping costly time integrations. Moreover, the Laplace response decouples the contributions from the tool and the environment. The first term in Eq. (18) encapsulates the impact of the moving heat source on the system modes, with the magnitude of the effect determined by the matrices $\mathbf{B}_r \mathbf{A}_{i_n}$, the decay of each mode from $e^{\lambda_m \tau}$, and the phase of the response by τ . Finally, the second and the third terms correspond to the response to the initial conditions and ambient environment.

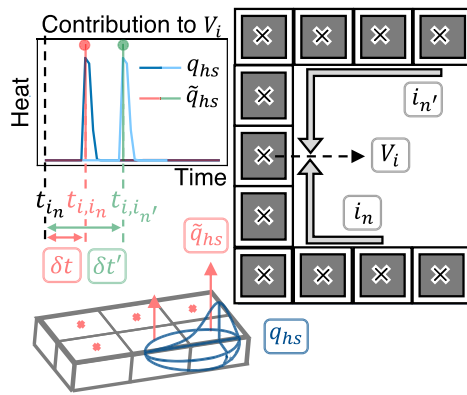


Fig. 2. Example of \tilde{q}_i and q_{hs} . The contribution of q_{hs} to each volume V_i is approximated using a series of Dirac pulses with amplitudes equal to the total energy transferred to V_i , during each move i_n in an arbitrary path I . This approximation enables the Laplace transform-based calculation of the steady-state response. The pulse phase solely depends on the path plan I .

The phase τ , uniquely defined for each node by the path plan I , effectively maps the members of the set J to temperature fields. For each control volume i in a move i_n starting at t_{i_n} , a corresponding entry t_{i,i_n} exists in t_{a,i_n} , calculated as

$$t_{i,i_n} = t_{i_n} + \delta t_{i,i_n}$$

where $\delta t_{i,i_n}$ is the time needed for the heat source to reach node i along the path of i_n . Consequently, every path $I = \{i_1, i_2, \dots, i_N\}$ results in a unique set of phase vectors $\{\tau_{i_1}, \tau_{i_2}, \dots, \tau_{i_N}\}$, derived as $\tau_{i_n} = t - t_{a,i_n} - nT$. This set of phases represents the only control available on the evolution of Eq. (18). An illustration of the phases can be found in Fig. 2.

Considering the periodic application of the path plan I , the steady-state response is also periodic. For large deposition periods T , the initial conditions settle before the end of a period, enabling the calculation of $v_m(nT) = v_m((n+1)T)$. This facilitates the efficient calculation of the steady-state response of the system. In this work, the contribution of the initial conditions converged to zero in the first 20% of the deposited layer, allowing the effect of the initial conditions to be calculated using the mode values at $v((n+1)T)$. For very short times T , this value can be approximated recursively.

3.4. Optimization

With Eq. (18), different paths I can be quickly evaluated, allowing the application of iterative optimization methods. Nonetheless, representing all possible permutations of a path plan is challenging and requires a structure that supports comparing similar choices. One way to achieve this is to represent each permissible path $I \in J$ as a branch of a state tree. State trees are mathematical structures that provide a powerful way to represent the state space of a decision-making problem, such as a game or, in this case, the sequence of the heat source moves. Each node within the state tree represents a particular feature-based move i_n , e.g. scanning feature f with the path f_j , and each edge represents a transition from one move to another. Thus, each path $I = \{i_1, i_2, \dots, i_N\}$ is represented by a branch with a certain state which defines the deposition order and the feature-specific paths as shown in Fig. 3.

3.4.1. Structure of a state tree

Each state tree has a *root* node, representing the deposition's initial state when no material has been added to a new layer. The root is then connected to its *children* nodes. The children of a node are all the nodes that can be reached by performing a single valid action, satisfying

the constraints of permissible paths. Similarly, a node's *parent* is the node from which this particular node originates. The set of all paths from the root to a node forms a sequence of actions, also known as a *policy* or *strategy*. In a state tree, any node that does not have children is associated with a *terminal* state S_T . All other nodes correspond to *non-terminal* states S_i . In the context of path planning, the terminal state represents a complete path I , which has covered the whole layer with material, while non-terminal states are parts of the layer path. An illustration of a state tree, annotated with its components can be found in Fig. 3.

3.4.2. State evaluation

Each terminal state is associated with a reward or a score, indicating the outcome of the associated path. Since a reward has positive connotations, following the optimization problem defined in Section 2, the reward \mathcal{V}_I associated with the path I is the negative of the maximum spatial standard deviation of the process-induced temperature fields occurring during a layer:

$$\mathcal{V}_I = - \left\| \text{std}(\theta(q_{hs}(t; I))) \right\|_{\infty} \quad (19)$$

over $t_0 \leq t \leq t_0 + T$

This score is used to evaluate the desirability of employing the path I .

State trees are integral to many artificial intelligence and machine learning algorithms, providing a structured framework to explore the state-action space of a problem exhaustively. However, for complex problems, the state tree can be prohibitively large, making it impossible to explore fully. In such scenarios, algorithms like Monte Carlo Tree Search can be employed to explore the state tree intelligently and efficiently, balancing the need for exploration and exploitation.

3.4.3. Monte Carlo tree search

Monte Carlo Tree Search (MCTS), originally proposed by [Kocsis and Szepesvári \(2006\)](#) and [Coulom \(2007\)](#), is an algorithmic approach largely employed in the domain of artificial intelligence for optimal decision-making in complex environments. It is particularly well-suited for problems with large state and action spaces, such as the game of Go ([Silver et al., 2016](#)) and chess ([Silver et al., 2018](#)), due to its stochastic, online, anytime nature. The advantage of MCTS when compared to exact methods such as MIP, is that while it only approximates the optimal choice of actions in a sequential decision-making problem, it can quickly rule out bad policies, focusing its evaluations on promising branches of the state tree. In this way, the algorithm provides near-optimal solutions within a limited time frame, even in problems with many and diverse moves. Consequently, MCTS allows for optimizing the path planning for parts regardless of their scale and complexity.

The MCTS algorithm consists of four primary phases: *Selection*, *Expansion*, *Simulation*, and *Backpropagation*.

Selection: The process begins from the tree's root, representing, in this case, the start of the deposition of a new layer. The algorithm traverses the tree in a depth-first approach, by recursively selecting and moving to one child node after the other until it reaches a non-terminal and not fully expanded node. The node selection process is typically performed using a tree policy. The Upper Confidence Bound applied to Trees (UCT) as introduced by [Coquelin and Munos \(2007\)](#) is a commonly used tree policy. It balances between exploration and exploitation by moving to the child node of a selected parent that maximizes the UCT formula

$$\text{UCT}(i) = \hat{\mathcal{V}}_i + C \sqrt{\frac{2 \ln \|S\|}{\|S_i\|}} \quad (20)$$

where $\hat{\mathcal{V}}_i$ is the average reward of the i th child node, $\|S\|$ is the number of times the current node has been

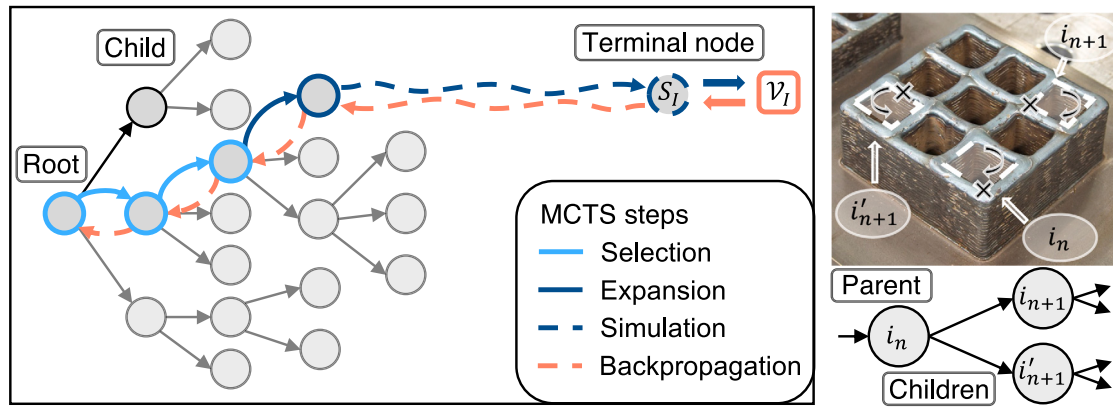


Fig. 3. A single iteration of Monte Carlo Tree Search (MCTS). This figure depicts a state tree, originating from the root node and branching out, highlighting the four essential steps of MCTS. Each branch represents the selection and ordering of various moves i_n that collectively form an admissible path I . In the simulation step, the reduced model is utilized to evaluate the response along the path from which the objective value is computed and backpropagated.

visited, $\|S_i\|$ is the number of times the i th child node has been visited, and C is an exploration constant balancing the urge to explore new branches of the state tree or exploit the accumulated experience. A typical choice for C is 1.41. In the event that multiple children have the same UCT score the algorithm selects one of the best alternatives at random.

Expansion: Once the selection phase reaches a node that is not fully expanded, the expansion phase generates one or more child nodes corresponding to valid actions. These newly created nodes are added to the tree.

Simulation: After expansion, a simulation (or rollout) is performed from the newly expanded node. The simulation follows a default (typically random) policy to rapidly reach a terminal node and create a state S_I , which is then evaluated using the ROM to return the reward \mathcal{V}_I .

Backpropagation: After the simulation phase, \mathcal{V}_I is backpropagated up the tree, updating the visit count $\|S\|$ and the cumulative reward $\hat{\mathcal{V}}$ of all nodes on the path from the new node to the root.

A graphical representation of these steps is given in Fig. 3.

The sequence of these four phases (selection, expansion, simulation, and backpropagation) is repeated for a given number of iterations or until a computational budget is exhausted. The action leading to the child node with the highest visit count from the root is then chosen as the best action.

Through this process, MCTS can concentrate its computational efforts on the most promising areas of the search space. At the same time, using the UCT tree policy, the algorithm retains the ability to explore the tree for other undiscovered auspicious candidates, thereby offering a favorable balance between exploitation and exploration.

To demonstrate the application of MCTS for path planning, we consider the part shown in Fig. 3, which depicts a thin-walled structure. The thin-walled geometry is discretized into ten distinct features, one for each pocket and one for the contour. Each feature accommodates multiple starting points and scanning directions, as highlighted in the figure with “x” and arrows. A dwell interval is introduced between subsequent layers, which is modeled as an additional feature and adds its duration to the layer length T .

The associated state tree for the paths of the part is formed by defining the states S_i of each node. In a given path $I = \{i_1, i_2, \dots, i_N\}$, the state S_n of the n th node corresponding to the i_n move is presented as an ordered set of tuples $S_n = \{(f_1, s_1, d_1), (f_2, s_2, d_2), \dots, (f_n, s_n, d_n)\}$, where f_n designates the scanned feature, s_n denotes the starting point on the contour, and d_n defines the scanning direction. The ranges

Table 1

Constant material properties of ER70S-6. T_s is the solidus temperature, T_L is the liquidus temperature, ρ is the density and, q_l is the latent heat.

Property	T_s [K]	T_L [K]	ρ [g/cm ³]	q_l [kJ/kg]
Value	1712	1766	7.8	270

for each of the f, s, d values are defined by the available moves. For instance, to describe all the moves of the part in Fig. 3, f_0 ranges from 0 to 10 including all pockets, the contour and the waiting time, s_n ranges from 0 to 7 and d_n takes either the value of -1 or 1 . The value of f_n depends on the previous choices of f , because a feature cannot be scanned twice.

Each node within the state tree represents a permissible move and a permissible path I is constructed by sequentially selecting nodes from the root, forming branches in the state tree. Each path I is directly associated with a terminal state $S_I = \{(f_1, s_1, d_1) \dots (f_N, s_N, d_N)\}$, resulting directly from the selected branch. Simulating the state S_I results in the reward \mathcal{V}_I which is then backpropagated to the tree. This structure, illustrated in Fig. 3, reveals that the order of the selected moves encapsulates the scanning sequence within each branch. By iteratively sampling branches according to the tree policy of MCTS, the algorithm can quickly identify near-optimal paths.

4. Experimental setup

To validate the proposed pipeline experimentally, a series of WAAM experiments are conducted using the setup illustrated in Fig. 4. The setup comprises a six-axis ABB IRB 1600 robot linked to a two-axis synchronized positioner. Welding is performed using the Fronius TPS 500i welding machine, utilizing a 1.2 mm diameter wire made of ER70S-6 structural steel. The assumed material properties are adopted directly from the work of Fabbri et al. (2024) and can be found in Tables 1 and 2.

The welding conditions involve a feed rate of 5 m/min, a welding speed of 400 mm/min, a nominal stick-out length of 15 mm, a layer height of 2.5 mm, a gas mixture of 82% Ar and 18% CO₂ resulting in 4.25 kW of generated thermal power based on the measurements of the Fronius welder and transferred to the part with an absorption coefficient equal to 0.85. To monitor the process, an Optris PI640 thermal camera is employed, which provides real-time thermal mapping ranging from -20°C to 900°C with a resolution of 75 mK and 640×480 pixels. Temperature measurements are calibrated using thermocouples to ensure accuracy. The system is controlled by the Robot Operating System (ROS), which runs on an external computer. The slicing, translation of paths into robot movements, data acquisition, and welding



Fig. 4. The welding cell used for the experimental validation of the pipeline.

Table 2

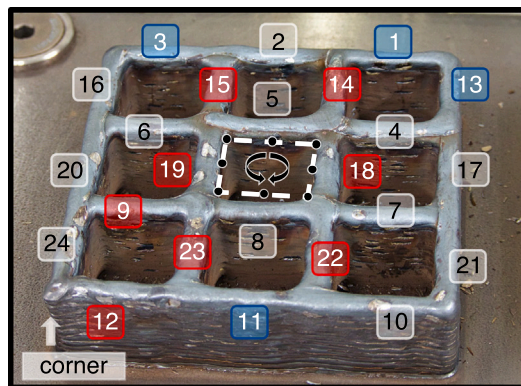
Temperature dependent material properties of ER70S-6. T is the temperature, K is the heat conductivity and, c_p is the heat capacity.

T [K]	293.15	373.15	473.15	573.15	693.15	773.15	873.15	973.15	1712	1766
K [W/(mK)]	51.9	51.1	48.6	44.4	42.7	39.4	35.6	31.8	31.8	250
c_p [J/(kg K)]	486	486	498	515	536	557	586	619	619	619

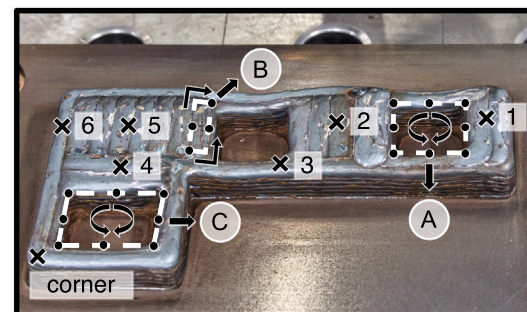
Table 3

Wall thicknesses. This table highlights the varying dimensions of the pocket walls when the initial path is employed. The color annotation corresponds to the wall IDs shown in Fig. 5(a).

Wall ID	1	3	11	13	9	12	14	15	18	19	22	23
Mean thickness [mm]	8.32	8.55	8.54	8.66	9.32	8.99	9.62	9.54	9.69	9.54	9.74	9.21
Standard Deviation [mm]	0.21	0.12	0.22	0.41	0.28	1.02	0.59	0.67	0.71	0.52	1.29	1.01



(a) Thin-walled pocket geometry



(b) Dense geometry

Fig. 5. Test geometries designed to highlight features susceptible to overheating. Figure (a) showcases a thin-walled structure, with labeled walls and color-coded annotations representing their respective thicknesses (as reported in Table 3). Each pocket within the structure can be filled using any permutation of the strategy depicted in the center of the component. This implies that the deposition can begin from any designated starting point, denoted by a dot, and proceed in any direction. Additionally, Figure (b) illustrates the three sets of moves employed to deposit the features of the dense geometry. The points where temperature measurements were recorded are also indicated by an x.

commands are based on an in-house Python framework developed on Compas (Mele et al., 2017-2021). This framework enables integrated control over the robot, welder, and thermal camera, to ensure effective coordination and streamlined operation.

The torch, which is continuously moving during deposition, interferes with the camera recordings, necessitating their post-processing. The post-processing is performed offline, pixel-wise and has 3 steps, namely, torch detection, removal and signal interpolation. To detect if the torch interferes with a pixel, the intensity history of each pixel

is examined, and wells of sudden intensity drop are identified. Then, these wells are removed from the history and are replaced with linearly interpolated values. Finally, the rectified signals are visually curated to guarantee that no artifacts were introduced.

For the experimental validation of the pipeline, the parts shown in Fig. 5 are designed. These geometries represent typical parts produced using deposition-based AM methods. Fig. 5(a) depicts a thin-walled structure with 9 mm walls, while Fig. 5(b) showcases a part combining three thin-walled pockets and three dense areas of varying thicknesses

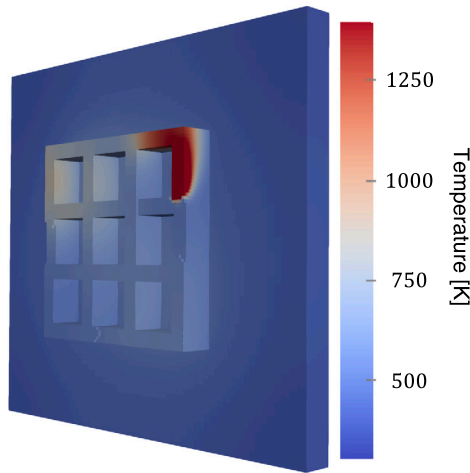


Fig. 6. Finite volume model temperature fields. The temperature fields showcased are derived from simulating a test geometry along a designated path. These fields correspond to a moment during the deposition process, when the current layer is not fully deposited. To enhance clarity in visualization, the maximum temperature is saturated at 1273.15 K.

(34 mm, 50 mm, and 82 mm). Both illustrated geometries were deposited using an intuitive path and failed at some point during the process due to large layer-height deviations, necessitating optimizing their deposition strategy.

Each thin wall consists of 2 weld tracks with a distance of 4 mm, while the dense parts are consolidated with rectangular equidistant tracks as demonstrated in Fig. 5(b). The thin-walled geometry is discretized into ten distinct features, one for each pocket and one for the contour, promoting a low frequency of start-stop actions per layer and ensuring comprehensive material coverage, thus minimizing deposition artifacts. Its dense counterpart comprises 16 features, which are split into 3 categories, A, B and C. Categories A and C model pockets of different sizes, and category B represents the building block of the dense parts. Each feature accommodates multiple starting points and scanning directions, as highlighted in Fig. 5. For both parts, a dwell interval of two minutes is introduced between subsequent layers, which is modeled as an additional feature to include its duration in the layer length T .

5. Results

The pipeline begins by meshing the geometries and simulating their thermal response to an intuitive path plan, using Abaqus for meshing and the FVM solver presented by Fabbri et al. (2024) for simulation. Both geometries require approximately 125 000 elements to be meshed. Fig. 6 shows an example of the resulting thermal fields while Figs. 10(a) and 10(c) illustrate the original paths.

Following simulation, the temperature-dependent material properties are approximated, and the dynamical matrices of the systems, $A(t)$, $B(t)$ and $B_b(t)$, are reconstructed and using Eq. (8). By calculating the singular values of the matrix $A(t)$ after each layer, the timestep t^* is determined as the point where the singular values have converged to at least 98% of their final value. Fig. 7 illustrates the rate of convergence of the first 1000 singular values of the $A(t)$ matrix corresponding to the parts of Fig. 5(a). After 4814.5 s the convergence rate of all singular values is close to zero, which indicates that the principal system dynamics do not change with the addition of more layers. Therefore, the transition time is set to $t^* = 4814.5$ s, corresponding to the tenth layer, with the matrices A_* , B_* and B_{b*} being defined at that exact moment.

Having defined the transition time and the steady-state dynamics of the full-scale systems, the reduction steps described in Section 3.2 are applied to obtain the ROMs. Starting from the decomposition

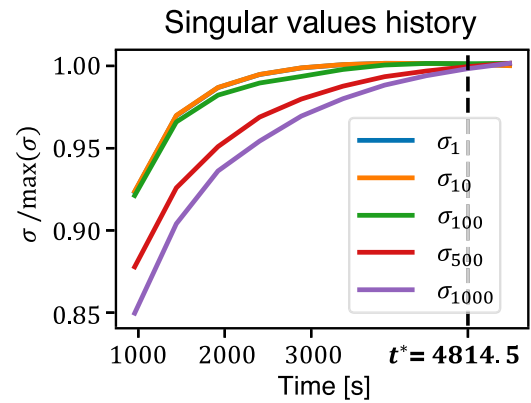


Fig. 7. Primary singular values of matrix A_* . Monitoring these singular values enables the determination of the system's transition time, which is defined at the point where the singular values converge.

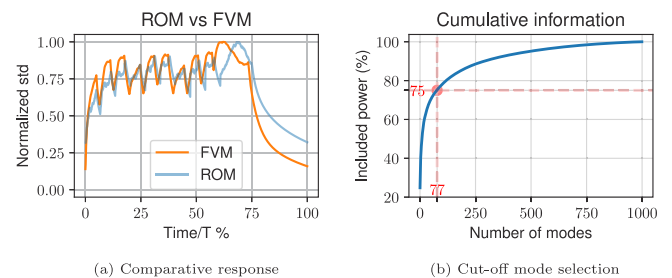


Fig. 8. Reduced order model (ROM) selection and performance. Figure (a) illustrates the comparison of responses between the original and the reduced system. Despite the reduced system's steady-state response being 10^4 orders of magnitude smaller than the original finite volume model, the ROM is able to follow the trend of the original response. Figure (b) showcases the total input-output energy retained in the ROM as the number of included modes r is selected at the elbow of the curve.

of the matrices A_* to their eigenpairs, the systems are diagonalized and truncated to their $M = 1000$ most prevalent dynamics through the transformation Z_{11} . For both systems, this transformation keeps dynamical modes whose eigenvalues span over 3 orders of magnitude. Then, by performing BT, the systems are further reduced to their r most important modes via the Φ^* transformation. For the parts of Fig. 5(a) this step results in an $r = 77$ degrees of freedom, while for the dense parts in Fig. 5(b) the corresponding value is $r = 76$. Finally, the ROM is diagonalized to obtain the matrix A_r and the transformation Z_3 reaching the representation of Eq. (14). The eigenvalues associated with the modes in the ROM range from -0.0013 to -0.3555 for the part in Fig. 5(a) and from -0.0010 to -0.4476 for Fig. 5(b).

Fig. 8(b) shows the cumulative amount of input-output power in the BT modes, $P_i = \frac{1}{\sum_{j=1}^M \sigma_j} \sum_{j=1}^i \sigma_j \times 100\%$, calculated for the part in Fig. 5(a). The elbow-like shape of the cumulative power curve demonstrates that a good trade-off between computational cost and expressiveness of the ROM can be achieved with $r = 77$ where the tangent to the curve is close to 45 degrees. After this point, including more information requires a rapidly increasing number of modes, making the system expensive to simulate.

The spatial standard deviation of the ROM's steady-state response closely aligns with the FVM simulation results, as shown in Fig. 8(a). Despite the ROM focusing on slowly decaying modes relevant to path planning, both signals exhibit similar trends, with truncated modes mainly affecting signal mean and scale. In terms of computational burden, the steady-state response in Eq. (18) is evaluated in approximately 0.04 s using an Nvidia 3090 GPU which is about $\times 12000$ faster than depositing a single layer. In comparison, simulating enough layers with the FVM solver to reach t^* takes about 2400 s using 5 cores of an intel

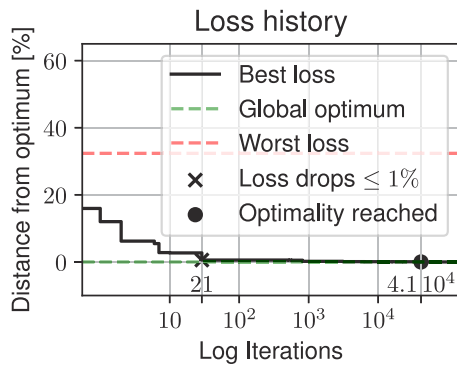


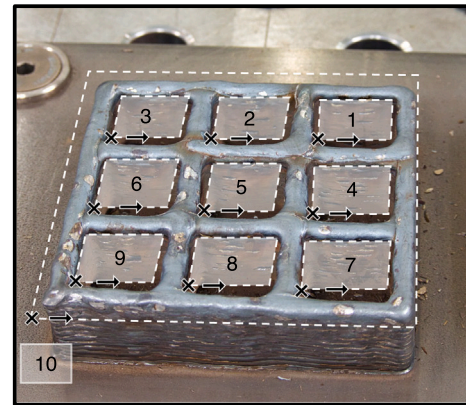
Fig. 9. Comparing Monte Carlo tree search (MCTS) with the optimal solution. For a simplified path planning problem, the best and worst paths were computed, and the solutions generated by the MCTS algorithm are compared to them. Remarkably, the algorithm is capable of swiftly obtaining a solution that is close to optimal, with the global optimum discovered within the initial 30 min of execution.

i7-8700K @ 3.70 GHz. This leads to a total time reduction factor of ×60000 for evaluating a single path relative to what would be needed without the steady-state response of the ROM. This speed-up is essential for the optimization algorithm since it allows iterating over hundreds of thousands of candidate paths in the span of a few hours.

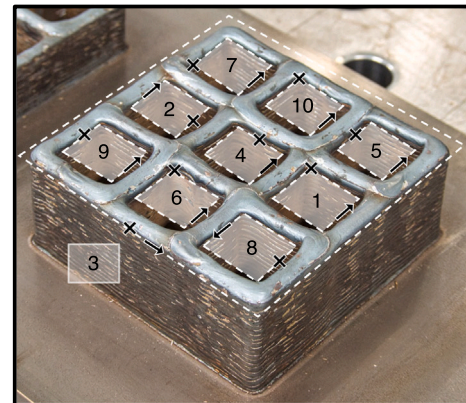
The next step to optimize the path is to split the geometries into distinct moves and model the state of the trees. Fig. 5, illustrates the distinct features of each geometry, together with the candidate feature-specific paths to consolidate them. The modeling of the state tree is analyzed in detail in Section 3.4.3. Moreover, examples of candidate moves can be also found in Fig. 10.

To test the MCTS' performance, a computational study is performed to compare its estimated optimal cost against the global optimum. To determine the global optimum, a full combinatorial search is performed testing all possible paths for the geometry in Fig. 5(a). However, to maintain the search computationally tractable, only the feature order is optimized, keeping the direction and starting point constant. This leaves 10! = 3.628.800 possible paths, which were computed in about 40 h. Then, the MCTS algorithm is run, again only optimizing the feature order. Fig. 9 demonstrates the evolution of the costs of the best-discovered strategies through the run. It is evident, that MCTS very quickly samples a near-optimal solution and within a few thousand iterations reaches the global optimum. The algorithm can find a solution that is 0.57% worse than the global optimum within 21 iterations, which corresponds to a time interval smaller than 1 s, and then reaches the global optimum in 41 000 iterations (≈27 min).

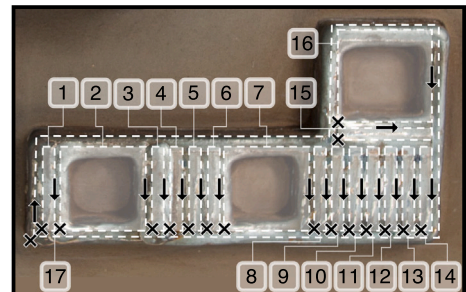
Next, MCTS is employed to discover the optimal paths for both geometries of Fig. 5. For this search, the candidate paths combine all possible features orders, starting points, and directions. The optimization algorithm is run for a total of 150 K iterations for the pocket part and 250 K for the dense geometry. A comparison between the initial and optimized paths can be found in Fig. 10. In contrast to the rather simple original strategies, where all the features were scanned in order and with the same starting point and direction, the optimal paths shuffle the feature order, starting points and scanning directions. This variation is a consequence of the loss function in Eq. (3), which seeks to minimize the maximum value of the spatial standard deviation during the deposition of a layer. The objective favors paths that evenly distribute the thermal load over the deposition plane to raise the minimum temperatures of the field and allow the warmer areas to cool. However, since the terminal states of the state trees, S_I , encode the total path of the layer, the algorithm is also able to optimally decide when to sacrifice some performance to avoid a combination of moves that would be detrimental to the temperature fields. To better compare with the initial solutions, the optimized paths are simulated on the FVM solver and the respective temperature fields are derived. Afterwards,



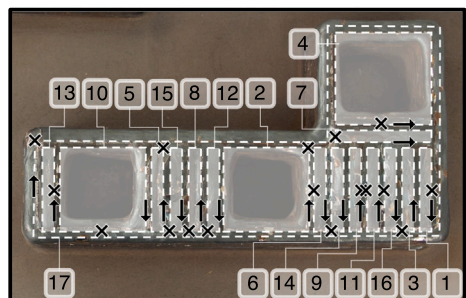
(a) Initial pocket path



(b) Optimal pocket path



(c) Initial dense path



(d) Optimal dense path

Fig. 10. Comparison of initial and optimal paths. This figure showcases the original and optimized paths for both geometries. The starting point of each feature's path is denoted with an "x", the direction with an arrow, and the order with a number. The optimized paths tend to rearrange the orders of features and evenly distribute the input heat across the deposition area, leading to a reduction in temperature standard deviation.

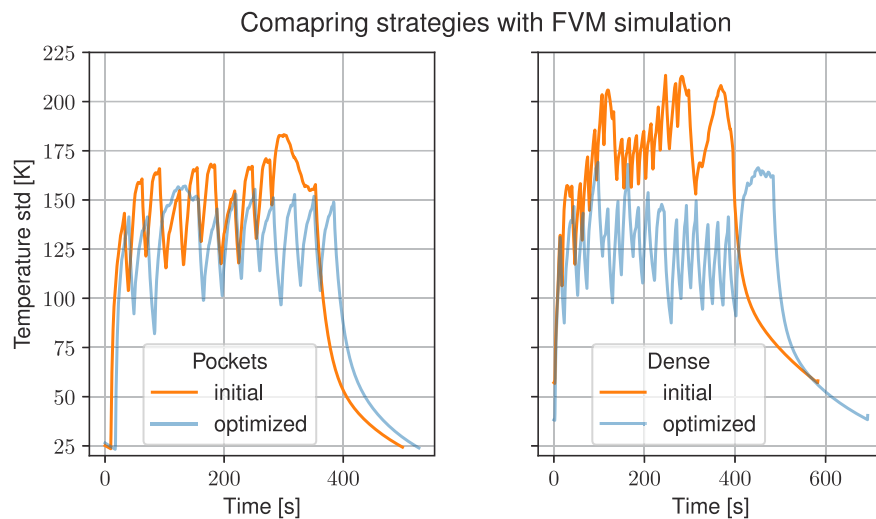


Fig. 11. Comparison between initial and optimal strategies. The optimized path is simulated using the finite volume method (FVM) model, and the spatial standard deviation is calculated and compared to the initial strategy. The plot on the left displays the standard deviation signals for the parts shown in Fig. 12(a), while the plot on the right pertains to Fig. 12(b). The optimized paths exhibit a noteworthy reduction in both the mean and maximum values of the signals.

Table 4

Performance improvement observed in the experimental and computational studies illustrating the potential of the proposed pipeline.

Improvement %	Simulation	Experiment
Pockets	14.00	25.49
L shape	19.70	33.94

their spatial standard deviation at the deposition layer is calculated for every timestep. Fig. 11 shows the resulting standard deviation signals of both geometries, for a period after t^* . The optimized paths result in a significantly reduced standard deviation profile, and the calculated improvement of the objective in Eq. (3) can be found in Table 4. This performance improvement can be primarily attributed to the better distribution of molten material over the deposition surface, and secondarily, to the longer deposition period, stemming from the dead time between the translation of the torch between different welding moves.

Further validating the performance of the pipeline, the original intuitive and optimized paths are fabricated for both geometries. The fabricated parts are shown in Fig. 12, where each row of figures shows the initial and optimized path. In both cases, the effect of the optimization is visible even by simple inspection. The parts created with the intuitive paths are characterized by layers with large height deviations due to material overflow. The deviations are most prevalent in the junctions of the pocket part and on the walls of the dense part, especially close to the concentrated mass. This results in the process being abruptly terminated within 14 layers for the pocket structure and 6 layers for the dense one. The cause is that due to the height inhomogeneity, the wire stick-out length greatly deviates, and the welder is not able to provide the demanded current to continue the process. By contrast, the optimized strategies can continue welding until we decide to stop them, depositing more than 20 layers of material.

Furthermore, the deviation in height also results in large deviations in the wall thickness, especially in the case of the pocket structure. Each wall of the structure is measured at three different locations and the histogram of all thicknesses for the optimal and the initial strategy are demonstrated in Fig. 13. For the optimal case, the wall thickness forms a narrow Gaussian distribution centered around 9 mm with a standard deviation approximating 0.15 mm. The walls of the initial strategy, though, follow a different trend. In the latter case, many outliers can be observed close to 8 mm and from 10 mm and above. To understand which walls correspond to the outliers, Table 3 lists all the walls with a mean thickness smaller than 8.7, larger than 9.3, or with a standard

deviation larger than 1 mm. The color annotations also correspond to the wall ID labels shown in Fig. 5(a).

To investigate the inconsistency in the wall thickness more closely, the recorded thermal histories of the defected walls are analyzed. Fig. 14 illustrates the thermal histories of some of them in comparison to their healthy-looking counterparts' that are produced with the optimal path. While the thermal camera saturates at 1173 K and does not allow the maximum temperatures reached during the process to be observed, comparing the initial and the optimized path shows that in the former case, the walls remained at a very high temperature for a long time compared to the optimal path. This is due to the subsequent passes of the torch over the same walls, leading to a welding process with a high preheat temperature, which changed the weld bead shape and gave rise to inconsistencies in the part dimensions. Nevertheless, this was not the case for the optimized path, where the distribution of moves around the plate led to significantly lower preheat temperatures and more repeatable welds.

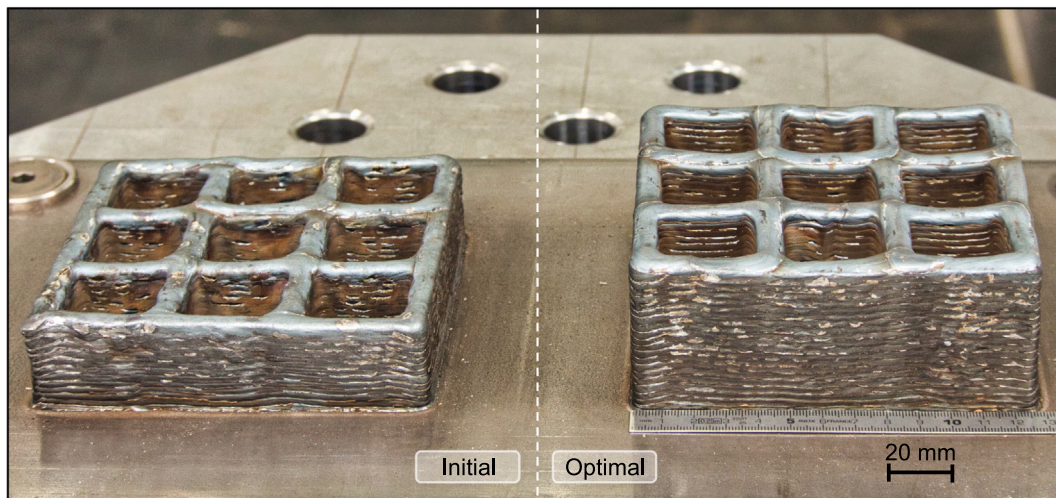
Similar conclusions can be drawn for the dense geometry, whose recorded thermal histories are found in Fig. 15 and they correspond to the points annotated in Fig. 5(b). Points 4, 5, and 6 remain at high temperatures for a substantial part of the deposition period since heat accumulates in this region and this leads to a high preheat temperature for the surrounding walls which collapse. This problem does not occur when the optimized path is used, which can be also observed in Fig. 12(b) where the inconsistency in height is mitigated.

Finally, the total estimated improvement of the objective in Eq. (3) for both parts can be found in Table 4. It is worth noting that the estimated improvement in the experiments is larger than expected from the simulations. The mismatch is mostly attributed to the boundary conditions encoded in the FVM model, which allow more energy to flow out of the system compared to reality. In practice, it is very challenging to exactly model the boundary conditions for WAAM, since the building plate is usually distorted during the process, and loses contact with the building platform, resulting in a smaller coefficient of heat transfer. As a result, the mean temperature of the parts is higher than predicted from the simulation, which also increases the importance of the optimized paths.

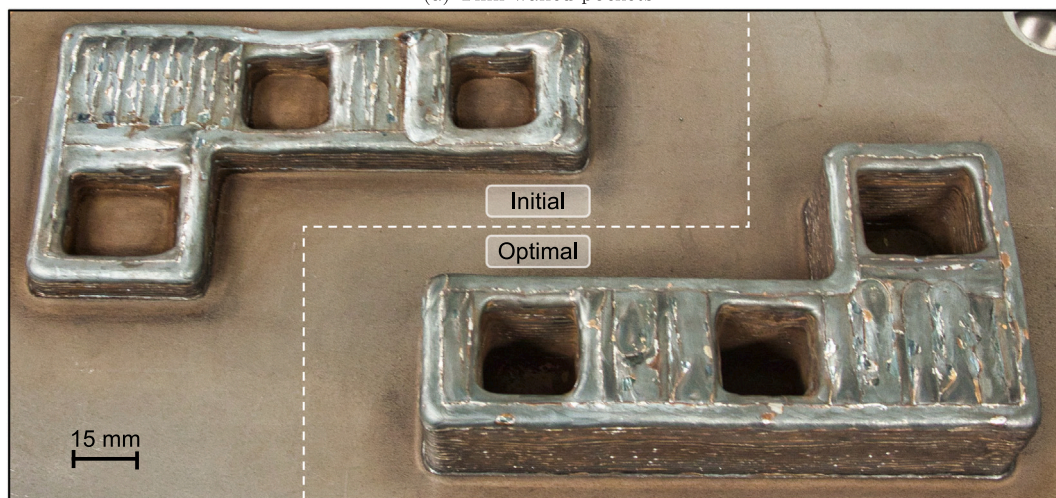
Segments of the thermal videos recorded during all four experiments can be found in the supplementary material.

6. Discussion

The progression of path planning in deposition-based additive manufacturing has been a journey of ongoing refinement, transitioning from



(a) Thin-walled pockets



(b) Geometry with dense areas

Fig. 12. Comparison between achieved geometries. The optimal paths result in higher quality parts, characterized by uniform dimensions and the absence of deposition artifacts. In contrast, the initial paths encountered challenges in completing the deposition process due to height changes within the layer.

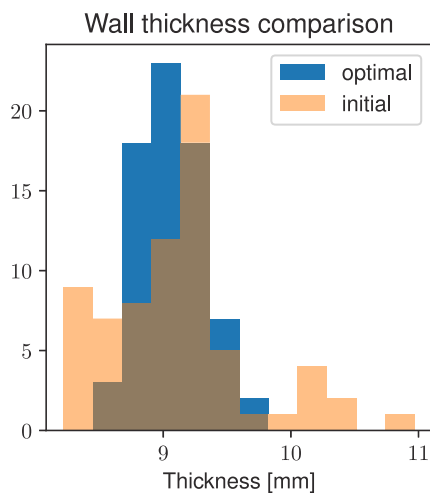


Fig. 13. Pockets wall thickness distribution. This histogram illustrates the empirical distributions of wall thicknesses generated using the initial and the optimal pocket deposition strategy. The optimal strategy generates walls with consistent thickness, whereas the initial strategy leads to a broad distribution of dimensions, adversely affecting part quality.

basic rules of thumb to intricate algorithms and heuristics. The primary objectives of the majority of the literature are to enhance the geometrical precision of the deposited parts, boost productivity, and mitigate defects. However, the findings of this study suggest that all these objectives are interconnected with the process-induced temperature fields, which are often overlooked.

6.1. Facing computational intractability

Incorporating the temperature fields as a variable in path planning significantly increases the complexity of the problem. This is primarily due to the intertwined relationship of temperature and shape, where predicting temperature is a challenging and computationally demanding task. In addition, the combinatorial nature of selecting and ordering the moves that optimize the fields has rendered temperature homogenization infeasible in previous studies, restricting their applicability to simplistic or toy examples.

This study addresses these issues directly, and intractability is effectively eliminated in four steps. First, by simplifying the thermal FVM model to its core constituents, modeling the heat input as an analytical function that depends solely on the path, and finding the steady-state response of the reduced system to this function, the computational burden of estimating the temperature fields is shifted. This reduction essentially allows for evaluating the response of a significantly smaller

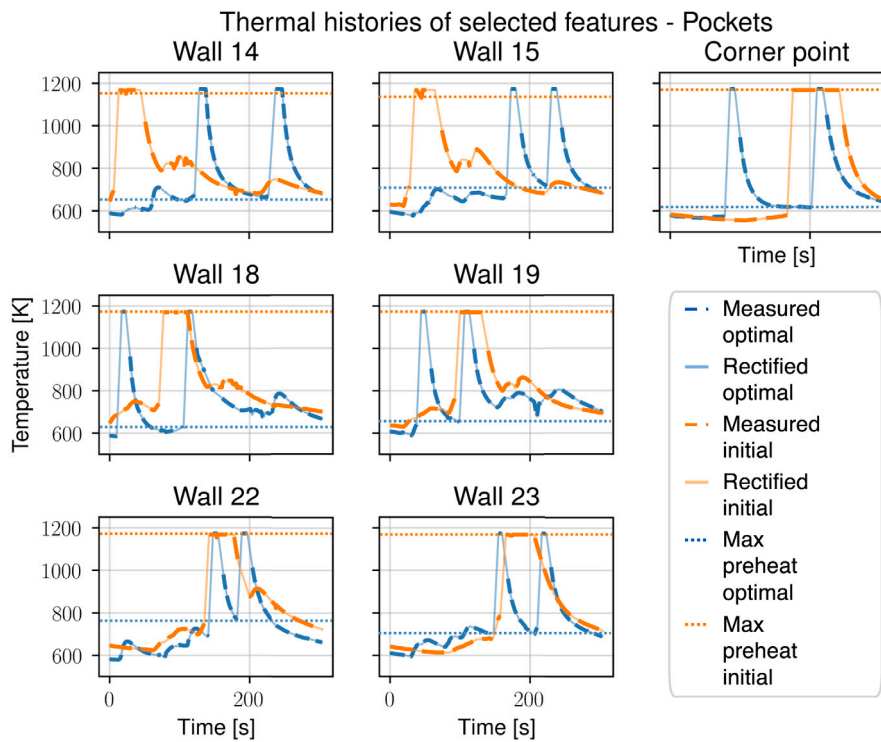


Fig. 14. Experimental thermal histories for the pocket structure. The thermal images undergo post-processing to eliminate the interference from the welding torch. In the original path, the welding preheat temperature consistently remains much higher compared to that of the optimal paths, leading to discrepancies in wall thickness.

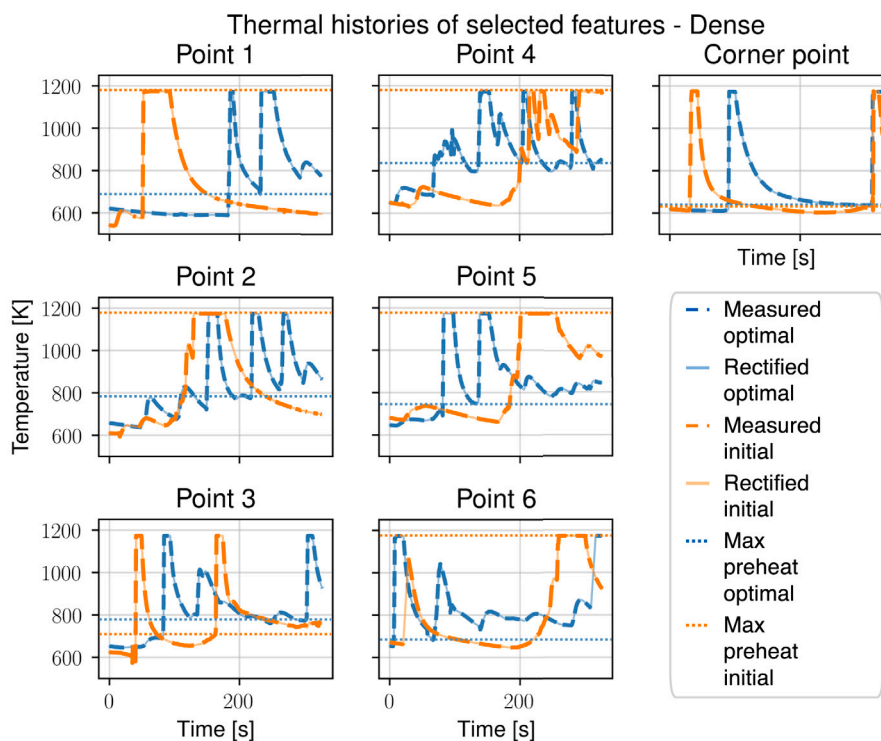


Fig. 15. Experimental thermal histories for the geometries with dense features. The thermal images undergo post-processing to eliminate the interference from the welding torch. In the original path, the dense areas concentrate heat, resulting in high preheat welding temperatures for the surrounding walls. This excessive heat causes the walls to eventually collapse. However, the optimal path effectively mitigates heat accumulation, resulting in improved wall quality.

system, compared to the original model, without the costly time integrations that FVM and FEM models require. Second, expressing the combinatorial optimization as a sequential decision-making problem and then approximating its solution with MCTS limits the number

of path evaluations compared to MIP as suggested by Beisegel et al. (2021), while retaining the ability to discover optimal solutions since it does not rely on heuristics or myopic selection of moves similar to Alhakeem et al. (2023). In fact, Fig. 9 demonstrates that MCTS was able

to find a solution 0.57, % close to the optimal path in only 21 iterations and reach optimality in about 41 000, while for the same geometry and deposition strategy (Beisegel et al., 2021) report hundreds of thousands. This showcases that MCTS is capable of handling complex geometries, thereby significantly enhancing the practicality and applicability of our method.

6.2. Simple and efficient implementation

One of the key advantages of the proposed optimization pipeline is its readiness for implementation. Unlike current literature that often requires the development of complex, domain-specific algorithms, the present solution is based on off-the-shelf components. This makes it easily adaptable to different part geometries and processes. Executing the pipeline only requires a system of ODEs, such as the weak formulation of the heat equation in Eq. (10), which can be obtained by the finite element method, the finite volume method, etc. Also, the modeling of the state tree is kept generic, and it could be easily reapplied to other processes by only changing the variables of the state tuple in each leaf. At the same time, BT is a standard technique for model reduction, widely used in control systems, while MCTS has been very popular in recent years due to its applications in reinforcement learning. All these algorithms are already implemented in several commercially available and open-source tools and cut down the pipeline's implementation effort.

The computational efficiency of the proposed method is another significant advantage that warrants discussion. The algorithm is designed to be efficient, with each process component optimized for speed. A typical simulation run takes just under an hour, the reduction process about an hour, and the MCTS approximately 2-3 h. This efficiency is achieved without requiring time-consuming and costly experiments to define a model, bringing the process closer to a first-time-right approach. Also, a single simulation is enough to optimize the path, keeping the computational overhead of the approach low. The computational costs of the method are primarily dependent on the mesh size, the level of compression in the ROM, and the number of permissible paths available. However, it is essential to note that MCTS has been proven to scale up to very large sets of possible moves. Specifically, the part of Fig. 12(a) can be deposited with approximately 10^{19} paths, while its counterpart in Fig. 12(b) has about 10^{30} candidate strategies. Nonetheless, the algorithm was able to handle both cases within the given time frame, even if the search space was scaled up by more than 10 orders of magnitude. This scalability demonstrates the versatility of the proposed method, allowing it to handle complex geometries without a significant increase in computational cost. The efficiency, combined with the method's robustness and adaptability, makes it a powerful tool for path planning in deposition-based additive manufacturing.

6.3. Robustness against model mismatch

Another noteworthy advantage of the developed method is its robustness, delivering reliable results even in the face of large mismatches between the simulation and actual system (sim-to-real gap). The developed optimization method relies on a reduced model that only includes the most important dynamics of the original system to compare different paths. As a result, any deviations between the FVM model and the process have a limited effect on the performance of the algorithm, in contrast to other optimization approaches that are heavily reliant on the precision of their models. Moreover, the disparity between the nominal process model and the physical system tends to be more pronounced in higher-order dynamics because of modeling complexity and parameter uncertainty issues. Higher-order dynamics inherently involve more complex interactions that are linked to the non-linearities of the strong form of Eq. (4), and they tend to be captured in the faster modes of the system, associated with eigenvalues of larger magnitude.

Due to the increased complexity and non-linearity, it is more difficult to identify the material properties and boundary conditions that apply to these parts of the system. For instance, identifying the emissivity and the conduction of the melt pool is challenging compared to the conduction coefficient of the solid material. However, in the ROM developed in this work, the modes including higher-order dynamics were explicitly cut off at the first step of the reduction as their effect quickly decays and does not affect the long-term behavior of the system. This ensures the resilience of the suggested pipeline in the presence of model uncertainty.

6.4. Effect on mechanical properties

The impact of the proposed approach on mechanical properties could be also notable. Following Figs. 14 and 15, inhomogeneous temperature fields lead to varying preheat temperatures, which in turn result in changing solidification conditions. One effect of this variation can be observed in the melt pools, which exhibit different dimensions within the same layer, leading to geometric inaccuracies as illustrated in Fig. 13. Another potential implication involves the microstructure and, consequently, the mechanical properties of the part. Israr et al. (2021) show that regulating the welding preheat temperature of ER70S-6 leads to a finer microstructure, higher toughness and yield strength. The improvement in the field homogeneity found in Table 4, suggests that the optimized parts present reduced microstructural variability. Also, the lower preheat temperatures in Figs. 14 and 15 indicate an improvement in the mechanical properties. However, it is important to note that further experimental validation is necessary to fully assess the effect of the path planning algorithm on the microstructure of the produced parts.

6.5. Connection to the flexible job-shop scheduling problem

Selecting and arranging the consecutive moves of a path can be framed as a flexible job shop scheduling problem, which is inherently NP-hard. Given the complexity of typical industrial parts, exhaustively exploring all possible solutions becomes infeasible for non-trivial instances of the problem. While there exist algorithms designed to yield near-optimal solutions in polynomial time (Höhn and Jacobs, 2012), these are generally tailored for simpler cost functions and do not seem to fit this specific use case. An alternative approach involves leveraging genetic algorithms (Pezzella et al., 2008; Valente and Gonçalves, 2009) and employing heuristics like the 2-Opt method, which has demonstrated a good performance in a variety of practical applications (Bentley, 1992). The 2-Opt heuristic serves a dual purpose: it condenses the problem into a more manageable NP-hard version and bounds the gap between the optimal and the achieved solutions (Brodowsky et al., 2023). This suggests that rephrasing the problem in this manner could significantly enhance the efficiency of the path planning pipeline.

6.6. Future work

Despite the numerous advantages of the proposed method, it is important to highlight a critical aspect that requires careful attention. The selection of features is a pivotal factor for a successful build. The robustness of the proposed algorithm does not extend to rectifying a poor choice of features or local paths. This limitation was evident in the example of Fig. 12(b), where the B-type moves resulted in overfilling the dense parts, and a different zig-gag or hatching strategy could potentially improve the resulting quality. Consequently, the successful application of the proposed method requires meticulous consideration and selection of features and paths. While the algorithm is robust and adaptable, the initial setup and choice of parameters play a significant role in the outcome. Therefore, users must ensure that the chosen features and paths are conducive to successfully applying the proposed method.

While the proposed method has shown significant promise, it is important to acknowledge an inherent complexity that it does not fully address. The order of moves plays a crucial role in the shape mismatch between the target and the produced geometry. This is not solely due to the temperature distribution but also because of the resulting bead geometry. For instance, the part resulting from the optimal path in Fig. 12(b) has a pronounced waviness on the surface of its dense areas compared to the original strategy. This aspect adds another layer of complexity to the problem, underscoring that although advanced methods have been applied, path planning in deposition-based AM remains a complex challenge.

To further improve the control over the deposition conditions, the thermal imaging system could be extended with a Kalman filter predicting the temperatures at areas obscured by the deposition tool, allowing for a full-field temperature estimation. This could be used for detecting deviations between the nominal and real temperature models and taking corrective actions, either by improving the underlying models or by closing the loop and calculating correction moves. Jones et al. (2002) have performed similar work for spray-forming processes, and their results can be adapted to serve deposition-based AM.

While only 2.5D geometries were fabricated in the present work, the proposed method can be also applied to parts with a finite number of cross-sections. For such parts, the study needs to be repeated for each section. It is likely that, the proposed approach could also be applied to parts with cross-sections that exhibit some periodicity. In such cases, achieving a steady state would require a larger period to capture the geometry changes. Furthermore, the authors anticipate that scaling the cross-sections would not impact the effectiveness of the algorithm, as long as the proportional sizes of the features within them remain consistent. The proposed method prioritizes and selects deposition patterns by seeking the optimal combination, and the scale of the cross-section itself does not influence the relative order of the features being deposited. However, continuously changing sections with arbitrary variations present a challenge due to the model reduction process. In fact, it is not trivial to find a generalized approach to define the t^* for arbitrary geometries since there is no guarantee that the primary singular values of the part will converge. A potential solution could involve slicing the part at various heights and adapting the path accordingly or using machine learning for the reduction. However, such a study is beyond the scope of this article.

7. Conclusions

This paper highlights the importance of the often overlooked process-induced temperature fields in deposition-based additive manufacturing and suggests a robust and efficient optimization pipeline for path planning based on reduced order modeling (ROM) and Monte Carlo Tree Search (MCTS), capable of accommodating complex geometries. The following conclusions can be drawn from the study:

1. Reducing a Finite Volume Method (FVM) model to its basic constituents can significantly shift the computational burden associated with handling large systems, thereby making path planning more manageable and efficient to solve.
2. Using the Laplace transformation to obtain the temperature steady-state response of a system to its path allows for evaluating the process-induced temperature fields without resorting to expensive time integrations.
3. The complex combinatorial optimization inherent in path planning can be transformed into a sequential decision-making problem and approximated with MCTS. This transformation not only simplifies the problem but also retains the ability to discover optimal solutions, as it does not rely on heuristics or myopic selection of moves.
4. MCTS has proven to be an effective tool for finding fast near-optimal solutions and scaling to very large systems. In our study,

MCTS was able to find a solution close to the optimal path in only a few iterations and reach optimality in a reasonable amount of time, even for complex geometries.

5. The use of the ROM increases the robustness of the approach against the sim-to-real gap. By relying on a reduced model that only includes the most important dynamics of the original system to compare different paths, the proposed method delivers reliable results even in the face of significant discrepancies between the simulation and the real process.
6. While the proposed method has shown significant promise, further work is needed to optimize the feature-specific paths and the interplay of the overlapping bead geometries. The selection of features is a pivotal factor for a successful build, and the robustness of the proposed algorithm does not extend to rectifying a poor choice of features and local paths. Therefore, future research should focus on these areas to further enhance the effectiveness and applicability of the proposed method.

CRedit authorship contribution statement

Iason Sideris: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Stephen Duncan:** Writing – review & editing, Supervision, Methodology, Formal analysis, Conceptualization. **Maicol Fabbri:** Writing – review & editing, Software. **Francesco Crivelli:** Writing – review & editing, Supervision. **Mohamadreza Afrasiabi:** Writing – review & editing, Supervision. **Markus Bambach:** Writing – review & editing, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used ChatGPT to improve the readability and language of this article. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Appendix A. Model reduction

In this appendix section, we provide a detailed derivation of the reduced order model. The first step towards obtaining the ROM in Eq. (12) is to perform the eigenvalue decomposition of the real-valued matrix A_* . To benefit from a stable and fast decomposition, the matrix is forced to be symmetric by evaluating the $A_{*s} = 1/2(A_* + A_*^T)$. For a grid with gradually changing cell dimensions, $A_* \approx A_{*s}$. This results in a set of orthogonal eigenvectors and eigenvalues that diagonalize A_{*s}

$$A_{*s} = Z_1^T \Lambda Z_1 \quad (\text{A.1})$$

where the orthogonal matrix $Z_1 \in \mathcal{R}^{N \times N}$ contains the eigenvectors and $\Lambda \in \mathcal{R}^{N \times N}$ is a diagonal matrix with the eigenvalues. Given the computational stability of the FVM model, A_* negative definite, meaning that all eigenvalues are negative and real. The goal of the first reduction is to keep in the system the M slowest and most persistent modes, so the eigenvalues and their corresponding eigenvectors are sorted in ascending magnitude order and the first M pairs are selected:

$$\begin{aligned} A_M &= \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_M) \\ Z_{11} &= [\zeta_1, \zeta_2, \dots, \zeta_M] \\ v_{11} &= Z_{11}^T \theta \end{aligned} \quad (\text{A.2})$$

where $\mathbf{A}_M \in \mathcal{R}^{M \times M}$ has the first M rows and columns of \mathbf{A} , $\mathbf{Z}_{11} \in \mathcal{R}^{N \times M}$ has the first M eigenvectors in $\mathbf{Z}_1 = [\mathbf{Z}_{11}, \mathbf{Z}_{12}]$ and $\mathbf{v}_{11} \in \mathcal{R}^M$ is the reduced state. Substituting $\theta = [\mathbf{Z}_{11}, \mathbf{Z}_{12}]\mathbf{v}_1$ in Eq. (11) the reduced state dynamics are calculated

$$\begin{aligned} \dot{\mathbf{v}}_{11} &= \mathbf{A}_M \mathbf{v}_{11} + \mathbf{B}_{11} \mathbf{q}_{hs} + \mathbf{B}_{1b1} \theta_a \\ \mathbf{y} &= \mathbf{C}_{11} \mathbf{v}_{11} \end{aligned} \quad (\text{A.3})$$

where $\mathbf{B}_{11} = \mathbf{Z}_{11}^T \mathbf{B} \in \mathcal{R}^{M \times U}$ and $\mathbf{B}_{1b1} = \mathbf{Z}_{11}^T \mathbf{B}_b \in \mathcal{R}^M$ are the input matrices and $\mathbf{C}_{11} = \mathbf{Z}_{11} \in \mathcal{R}^{N \times M}$ is the measurement matrix. The $\mathbf{Z}_{12} \in \mathcal{R}^{N \times N-M}$ submatrix maps to all the discarded modes.

The next step in the model reduction pipeline is to perform BT (Safonov and Chiang, 1988) on Eq. (A.3). The balancing transformation $\mathbf{Z}_2 \in \mathcal{R}^{M \times M}$ of the system is a coordinate map that results in $\hat{\mathbf{W}}_c = \hat{\mathbf{W}}_o = \Sigma_H$, where $\mathbf{W}_c \in \mathcal{R}^{M \times M}$ and $\mathbf{W}_o \in \mathcal{R}^{M \times M}$ are the controllability and observability Gramians of the original system, the hat refers to the transformed coordinates and Σ_H includes the diagonal entries σ_i which are the Hankel singular values of the system. Each σ_i can be interpreted as the amount of energy tied to the i th mode in \mathbf{Z}_2 and the number r of σ_i included in the ROM dictates the proportion of input-output signal power seized by the reduction. Consequently, including r modes retains $P_r\%$ of the total energy included in Eq. (A.3), and this leads to an informed decision on the choice of r . An example of how r is chosen can be found in Fig. 8(b).

Applying BT involves computing the original \mathbf{W}_c and \mathbf{W}_o by solving the Lyapunov equations

$$\begin{aligned} \mathbf{A}_M \mathbf{W}_c + \mathbf{W}_c \mathbf{A}_M^T + \mathbf{B}_{11} \mathbf{B}_{11}^T &= 0 \\ \mathbf{A}_M^T \mathbf{W}_o + \mathbf{W}_o \mathbf{A}_M + \mathbf{C}_{11}^T \mathbf{C}_{11} &= 0 \end{aligned} \quad (\text{A.4})$$

with a variant of the Bartels–Stewart algorithm (Barraud, 1977; Bartels and Stewart, 1972; Hammarling, 1982). The $\mathbf{B}_{1b1} \theta_a$ term is constant and considered as a disturbance, thus not included in the \mathbf{W}_c calculations. Once \mathbf{W}_c and \mathbf{W}_o are computed, the following eigenvalue problem is solved to find a scaled version \mathbf{Z}_{2s} of \mathbf{Z}_2

$$\mathbf{W}_c \mathbf{W}_o \mathbf{Z}_{2s} = \mathbf{Z}_{2s} \Sigma_H^2 \quad (\text{A.5})$$

Then \mathbf{Z}_{2s} is rescaled as

$$\text{Step 1: } \Sigma_c = \mathbf{Z}_{2s}^{-1} \mathbf{W}_c \mathbf{Z}_{2s}^{-*}, \Sigma_o = \mathbf{Z}_{2s}^* \mathbf{W}_o \mathbf{Z}_{2s}$$

$$\text{Step 2: } \Sigma_s = \Sigma_c^{\frac{1}{4}} \Sigma_o^{-\frac{1}{4}}$$

$$\text{Step 3: } \mathbf{Z}_2 = \mathbf{Z}_{2s} \Sigma_s$$

Here $\Sigma_s \in \mathcal{R}^{M \times M}$ is just a scaling matrix.

Based on the decision of r , matrices Φ and Ψ are defined. Ψ is composed of the first r columns of \mathbf{Z}_2 , and Φ^* by the first r rows of \mathbf{Z}_2^{-1} . The truncated system can be expressed as

$$\begin{aligned} \dot{\mathbf{v}}_{2r} &= \mathbf{A}_{2r} \mathbf{v}_{2r} + \mathbf{B}_{2r} \mathbf{q}_{hs} + \mathbf{B}_{2br} \theta_a \\ \mathbf{y} &= \mathbf{C}_{2r} \mathbf{v}_{2r} \end{aligned} \quad (\text{A.6})$$

where $\mathbf{v}_{2r} = \Phi^* \mathbf{v}_{11} \in \mathcal{R}^r$ are the r most important modes for the input–output behavior of the system, $\mathbf{A}_{2r} = \Phi^* \mathbf{A}_M \Psi \in \mathcal{R}^{r \times r}$ expresses the ROM dynamics, $\mathbf{C}_{2r} = \mathbf{C}_{11} \Psi \in \mathcal{R}^{N \times r}$ is the measurement matrix, $\mathbf{B}_{2r} = \Phi^* \mathbf{B}_{11} \in \mathcal{R}^{r \times U}$ and, $\mathbf{B}_{2br} = \Phi^* \mathbf{B}_{1b1} \in \mathcal{R}^r$ are the matrices mapping the input heat and boundary conditions to the reduced state.

As a final step, to simplify the steady-state calculation in Section 3.3, Eq. (A.6) is diagonalized. Similar to Eqs. (A.1) and (A.5), the diagonalization is performed by solving the eigenvalue problem

$$\mathbf{A}_{2r} \mathbf{Z}_3 = \mathbf{Z}_3 \Lambda_r \quad (\text{A.7})$$

and by applying the transformation \mathbf{Z}_3 , Eq. (A.6) can be rewritten as

$$\begin{aligned} \dot{\mathbf{v}} &= \Lambda_r \mathbf{v} + \mathbf{B}_r \mathbf{q}_{hs} + \mathbf{B}_{rb} \theta_a \\ \mathbf{y} &= \mathbf{C}_r \mathbf{v} \end{aligned} \quad (\text{A.8})$$

where $\mathbf{v} = \mathbf{Z}_3^{-1} \mathbf{v}_{2r} \in \mathcal{R}^r$ are the modes of Eq. (A.6), $\mathbf{C}_r = \mathbf{C}_{2r} \mathbf{Z}_3 \in \mathcal{R}^{N \times r}$ is the measurement matrix, $\mathbf{B}_r = \mathbf{Z}_3^{-1} \mathbf{B}_{2r} \in \mathcal{R}^{r \times U}$ and, $\mathbf{B}_{rb} = \mathbf{Z}_3^{-1} \mathbf{B}_{2br} \in$

\mathcal{R}^r are the matrices mapping the input heat and boundary conditions to the reduced state. The total transformation from θ to \mathbf{v}_3 can then be expressed as

$$\begin{aligned} \mathbf{v} &= \mathbf{Z}_r \theta \\ \mathbf{Z}_r &= \mathbf{Z}_3^{-1} \Phi^* \mathbf{Z}_{11}^T \end{aligned} \quad (\text{A.9})$$

Appendix B. Steady-state response to path

Starting from the ROM in Eq. (14), applying the Laplace transform \mathcal{L} to the m th mode, v_m , we get

$$\begin{aligned} \dot{v}_m &= \lambda_m v_m + \mathbf{B}_{r,m} \tilde{\mathbf{q}}_{hs} + b_{rb,m} \theta_a \xrightarrow{\mathcal{L}} \\ sV_m - v_m(t = nT) &= \lambda_m V_m + \mathcal{L}\left\{ \mathbf{B}_{r,m} \tilde{\mathbf{q}}_{hs} \right\} + b_{rb,m} \theta_a \frac{1}{s} \Rightarrow \\ V_m &= \mathcal{L}\left\{ \mathbf{B}_{r,m} \tilde{\mathbf{q}}_{hs} \right\} \frac{1}{s - \lambda_m} + b_{rb,m} \theta_a \frac{1}{s} \frac{1}{s - \lambda_m} + v_m(nT) \frac{1}{s - \lambda_m} \end{aligned} \quad (\text{B.1})$$

where the subscript m defines the m th row of the corresponding matrices, and $\mathcal{L}\{v_m\} = V_m$.

Further analyzing the input term

$$\begin{aligned} \mathcal{L}\left\{ \mathbf{B}_{r,m} \tilde{\mathbf{q}}_{hs} \right\} &= \mathcal{L}\left\{ \sum_{i_n \in I} \mathbf{B}_{r,m} \mathbf{A}_{i_n} \Delta(\tau) \right\} \\ &= \sum_{i_n \in I} \mathbf{B}_{r,m} \mathbf{A}_{i_n} \mathcal{L}\left\{ \Delta(\tau) \right\} \\ &= \sum_{i_n \in I} \mathbf{B}_{r,m} \mathbf{A}_{i_n} E_{i_n}^s(s) \end{aligned} \quad (\text{B.2})$$

with the matrix $E_{i_n}^s(s)$ being the stack of $e_i = \exp(-(t_{i,i_n} + nT)s)$ for every node i

$$E_{i_n}^s = \begin{bmatrix} \exp(-(t_{1,i_n} + nT)s) \\ \exp(-(t_{2,i_n} + nT)s) \\ \vdots \\ \exp(-(t_{U,i_n} + nT)s) \end{bmatrix}$$

Finally, the inverse Laplace transform \mathcal{L}^{-1} is applied to Eq. (B.1) to calculate the steady-state response of Eq. (14) corresponding to the deposition path plan I . After decoupling the terms in the responses and crossing out transient terms, the response for a single mode v_m is given by the equation

$$v_m(t) = \sum_{i_n} \mathbf{B}_{r,m} \mathbf{A}_{i_n} (\mathbf{H}(\tau) \circ e^{\lambda_m \tau}) + v_m(nT) \exp(\lambda_m \tau_0) - \frac{b_{rb,m}}{\lambda_m} \theta_a \quad (\text{B.3})$$

In this equation, \circ is the Hadamard product, $\tau_0 = t - nT$ is the time spent on the current layer, and the vectors $\mathbf{H}(\tau) \in \mathcal{R}^U$ and $e^{\lambda_m \tau} \in \mathcal{R}^U$ fabricate the impulse response of each mode, and are given by

$$\mathbf{H}(\tau) = \begin{bmatrix} h(\tau_{1,i_n}) \\ h(\tau_{2,i_n}) \\ \vdots \\ h(\tau_{U,i_n}) \end{bmatrix}, e^{\lambda_m \tau} = \begin{bmatrix} \exp(\lambda_m \tau_{1,i_n}) \\ \exp(\lambda_m \tau_{2,i_n}) \\ \vdots \\ \exp(\lambda_m \tau_{U,i_n}) \end{bmatrix}$$

with $h(\tau_{i,i_n})$ being the Heaviside function activated at τ_{i,i_n} .

By further grouping the Eq. (B.3), we can reach the formulation

$$v_m(t) = \sum_{i_n} E_{i_n,m}(\tau) + E_{0,m}(\tau_0) - \frac{b_{rb,m}}{\lambda_m} \theta_a \quad (\text{B.4})$$

where $E_{i_n,m}(\tau) = \mathbf{B}_{r,m} \mathbf{A}_{i_n} (\mathbf{H}(\tau) \circ e^{\lambda_m \tau}) \in \mathcal{R}$ is the response to the heat source and $E_{0,m}(\tau_0)$ models the initial conditions at the start of the path. Finally, grouping the responses of all modes $m = 1 \dots r$ leads to Eq. (18).

Appendix C. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.jmatprotec.2024.118364>.

References

- Akram, J., Chalavadi, P., Pal, D., Stucker, B., 2018. Understanding grain evolution in additive manufacturing through modeling. *Addit. Manuf.* 21, 255–268. <http://dx.doi.org/10.1016/j.addma.2018.03.021>.
- Alhakeem, M.M., Mollamahmutoglu, M., Yilmaz, O., Bol, N., Kara, O.E., 2023. A deposition strategy for wire arc additive manufacturing based on temperature variance analysis to minimize overflow and distortion. *J. Manuf. Process.* 85, 1208–1220. <http://dx.doi.org/10.1016/j.jmapro.2022.11.006>.
- Barraud, A., 1977. A numerical algorithm to solve $AT X A - X=Q$. In: 1977 IEEE Conference on Decision and Control Including the 16th Symposium on Adaptive Processes and a Special Symposium on Fuzzy Set Theory and Applications. IEEE, New Orleans, LA, USA, pp. 420–423. <http://dx.doi.org/10.1109/CDC.1977.271607>.
- Bartels, R.H., Stewart, G.W., 1972. Algorithm 432 [C2]: Solution of the matrix equation $AX + XB = C$ [F4]. *Commun. ACM* 15 (9), 820–826. <http://dx.doi.org/10.1145/361573.361582>.
- Beisegel, J., Buhl, J., Israr, R., Schmidt, J., Bambach, M., Fügenschuh, A., 2021. Mixed-integer programming for additive manufacturing. <http://dx.doi.org/10.26127/BTUOpen-5731>.
- Bentley, J.J., 1992. Fast algorithms for geometric traveling salesman problems. *ORSA J. Comput.* 4 (4), 387–411. <http://dx.doi.org/10.1287/ijoc.4.4.387>.
- Brodowsky, U.A., Hougardy, S., Zhong, X., 2023. The approximation ratio of the k -opt heuristic for the Euclidean traveling salesman problem. *SIAM J. Comput.* 52 (4), 841–864. <http://dx.doi.org/10.1137/21M146199X>.
- Brunton, S.L., Kutz, J.N., 2019. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, pp. 321–330. <http://dx.doi.org/10.1017/9781108380690>.
- Bui, H., Pierson, H.A., Nurre, S.G., Sullivan, K.M., 2019. Tool path planning optimization for multi-tool additive manufacturing. *Procedia Manuf.* 39, 457–464. <http://dx.doi.org/10.1016/j.promfg.2020.01.389>.
- Coquelin, P.-A., Munos, R., 2007. Bandit algorithms for tree search. [arXiv:cs/0703062](https://arxiv.org/abs/cs/0703062).
- Couloum, R., 2007. Efficient selectivity and backup operators in Monte-Carlo tree search. In: van den Herik, H.J., Ciancarini, P., Donkers, H.H.L.M.J. (Eds.), *Computers and Games*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 72–83.
- Diouré, A., Bugarin, F., Bordreuil, C., Segonds, S., 2021. Continuous three-dimensional path planning (CTPP) for complex thin parts with wire arc additive manufacturing. *Addit. Manuf.* 37, 101622. <http://dx.doi.org/10.1016/j.addma.2020.101622>.
- Fabbri, M., Aschwanden, I., Wegener, K., Bambach, M., 2024. A finite volume scheme with an adaptive heat flux calculation for fast temperature prediction for wire arc additive manufacturing. *J. Mater. Process. Technol.* 324, 118281. <http://dx.doi.org/10.1016/j.jmatprotec.2023.118281>.
- Foroozmehr, E., Kovacevic, R., 2010. Effect of path planning on the laser powder deposition process: thermal and structural evaluation. *Int. J. Adv. Manuf. Technol.* 51 (5–8), 659–669. <http://dx.doi.org/10.1007/s00170-010-2659-6>.
- Goldak, J., Chakravarti, A., Bibby, M., 1984. A new finite element model for welding heat sources. *Metall. Trans. B* 15 (2), 299–305. <http://dx.doi.org/10.1007/BF02667333>.
- Hammarling, S.J., 1982. Numerical solution of the stable, non-negative definite Lyapunov equation. *IMA J. Numer. Anal.* 2 (3), 303–323. <http://dx.doi.org/10.1093/imanum/2.3.303>, eprint: <https://academic.oup.com/imanj/article-pdf/2/3/303/2293676/2-3-303.pdf>.
- Höhn, W., Jacobs, T., 2012. An experimental and analytical study of order constraints for single machine scheduling with quadratic cost. In: Bader, D.A., Mutzel, D. (Eds.), 2012 Proceedings of the Fourteenth Workshop on Algorithm Engineering and Experiments. ALENEX, Society for Industrial and Applied Mathematics, Philadelphia, PA, pp. 103–117. <http://dx.doi.org/10.1137/1.9781611972924.11>.
- Israr, R., Buhl, J., Bambach, M., 2021. A study on power-controlled wire-arc additive manufacturing using a data-driven surrogate model. *Int. J. Adv. Manuf. Technol.*
- Johnson, S.M., 1954. Optimal two- and three-stage production schedules with setup times included. *Nav. Res. Logist. Q.* 1 (1), 61–68. <http://dx.doi.org/10.1002/nav.3800010110>.
- Jones, P., Pathirana, P., Duncan, S., 2002. Observer design for control of the spray-forming of tooling. *IFAC Proc. Vol.* 35 (1), 139–144. <http://dx.doi.org/10.3182/20020721-6-ES-1901.00026>.
- Kocsis, L., Szepesvári, C., 2006. Bandit based Monte-Carlo planning. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (Eds.), *Machine Learning: ECML 2006*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 282–293.
- Lall, S., Marsden, J.E., Glavaški, S., 2002. A subspace approach to balanced truncation for model reduction of nonlinear control systems. *Internat. J. Robust Nonlinear Control* 12 (6), 519–535. <http://dx.doi.org/10.1002/rnc.657>.
- Lawler, E.L., Lenstra, J.K., Kan, A.H.G.R., Shmoys, D.B., 1993. Chapter 9 sequencing and scheduling: Algorithms and complexity. In: *Logistics of Production and Inventory*. In: *Handbooks in Operations Research and Management Science*, vol. 4, Elsevier, pp. 445–522. [http://dx.doi.org/10.1016/S0927-0507\(05\)80189-6](http://dx.doi.org/10.1016/S0927-0507(05)80189-6), ISSN: 0927-0507.
- Li, C., Liu, Z., Fang, X., Guo, Y., 2018. Residual stress in metal additive manufacturing. *Procedia CIRP* 71, 348–353. <http://dx.doi.org/10.1016/j.procir.2018.05.039>.
- Mele, T.V., et al., 2017–2021. COMPAS: A framework for computational research in architecture and structures. <http://dx.doi.org/10.5281/zenodo.2594510>, <http://compas.dev>.
- Michel, F., Lockett, H., Ding, J., Martina, F., Marinelli, G., Williams, S., 2019. A modular path planning solution for wire + arc additive manufacturing. *Robot. Comput.-Integr. Manuf.* 60, 1–11. <http://dx.doi.org/10.1016/j.rcim.2019.05.009>.
- Moore, B., 1981. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Trans. Automat. Control* 26 (1), 17–32. <http://dx.doi.org/10.1109/TAC.1981.1102568>.
- Mozaffar, M., Liao, S., Lin, H., Ehmann, K., Cao, J., 2021. Geometry-agnostic data-driven thermal modeling of additive manufacturing processes using graph neural networks. *Addit. Manuf.* 48, 102449. <http://dx.doi.org/10.1016/j.addma.2021.102449>.
- Nguyen, L., Buhl, J., Bambach, M., 2020. Continuous Eulerian tool path strategies for wire-arc additive manufacturing of rib-web structures with machine-learning-based adaptive void filling. *Addit. Manuf.* 35, 101265. <http://dx.doi.org/10.1016/j.addma.2020.101265>.
- Ogata, K., 2010. *Modern Control Engineering*. In: *Instrumentation and controls series*, Prentice Hall, pp. 15–17.
- Petrik, J., Bambach, M., 2023. Reinforcement learning and optimization based path planning for thin-walled structures in wire arc additive manufacturing. *J. Manuf. Process.* 93, 75–89. <http://dx.doi.org/10.1016/j.jmapro.2023.03.013>.
- Pezzella, F., Morganti, G., Ciaschetti, G., 2008. A genetic algorithm for the flexible job-shop scheduling problem. *Comput. Oper. Res.* 35 (10), 3202–3212. <http://dx.doi.org/10.1016/j.cor.2007.02.014>.
- Safonov, M.G., Chiang, R.Y., 1988. A Schur method for balanced model reduction. In: 1988 American Control Conference. pp. 1036–1040. <http://dx.doi.org/10.23919/ACC.1988.4789873>.
- Ščetinec, A., Klobčar, D., Bračun, D., 2021. In-process path replanning and online layer height control through deposition arc current for gas metal arc based additive manufacturing. *J. Manuf. Process.* 64, 1169–1179. <http://dx.doi.org/10.1016/j.jmapro.2021.02.038>.
- Schmitz, M., Wiartalla, J., Gelfgren, M., Mann, S., Corves, B., Hüsing, M., 2021. A robot-centered path-planning algorithm for multidirectional additive manufacturing for WAAM processes and pure object manipulation. *Appl. Sci.* 11 (13), 5759. <http://dx.doi.org/10.3390/app11135759>.
- Sideris, I., Crivelli, F., Bambach, M., 2023. GPyro: uncertainty-aware temperature predictions for additive manufacturing. *J. Intell. Manuf.* 34 (1), 243–259. <http://dx.doi.org/10.1007/s10845-022-02019-7>.
- Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D., 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529 (7587), 484–489. <http://dx.doi.org/10.1038/nature16961>.
- Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., Hassabis, D., 2018. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science* 362 (6419), 1140–1144. <http://dx.doi.org/10.1126/science.aar6404>.
- Valente, J.M., Gonçalves, J.F., 2009. A genetic algorithm approach for the single machine scheduling problem with linear earliness and quadratic tardiness penalties. *Comput. Oper. Res.* 36 (10), 2707–2715. <http://dx.doi.org/10.1016/j.cor.2008.11.016>.
- Venturini, G., Montevicchi, F., Scippa, A., Campatelli, G., 2016. Optimization of WAAM deposition patterns for T-crossing features. *Procedia CIRP* 55, 95–100. <http://dx.doi.org/10.1016/j.procir.2016.08.043>.
- Xia, L., Bi, M., Wu, J., Wang, F., Wang, L., Xie, Y.M., Ma, G., 2023. Integrated lightweight design method via structural optimization and path planning for material extrusion. *Addit. Manuf.* 62, 103387. <http://dx.doi.org/10.1016/j.addma.2022.103387>.
- Xia, L., Lin, S., Ma, G., 2020. Stress-based tool-path planning methodology for fused filament fabrication. *Addit. Manuf.* 32, 101020. <http://dx.doi.org/10.1016/j.addma.2019.101020>.
- Xiong, H., Shi, S., Ren, D., Hu, J., 2022. A survey of job shop scheduling problem: The types and models. *Comput. Oper. Res.* 142, 105731. <http://dx.doi.org/10.1016/j.cor.2022.105731>.
- Yamamoto, K., Luces, J.V.S., Shirasu, K., Hoshikawa, Y., Okabe, T., Hirata, Y., 2022. A novel single-stroke path planning algorithm for 3D printers using continuous carbon fiber reinforced thermoplastics. *Addit. Manuf.* 55, 102816. <http://dx.doi.org/10.1016/j.addma.2022.102816>.
- Yang, Y., Billingham, J., Axinte, D., Liao, Z., 2023. A rational approach to beam path planning in additive manufacturing: the inverse heat placement problem. *Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci.* 479 (2270), 20220386. <http://dx.doi.org/10.1098/rspa.2022.0386>.