Predicting Temperature Field for Metal Additive Manufacturing using PINN

B. Peng, A. Panesar

IDEA Lab, Department of Aeronautics, Imperial College London,
London, United Kingdom

Abstract

Machine-learning-based methods are gaining traction as an alternative to numerical methods in many engineering applications. Physics-informed neural network (PINN), a self-supervised method, is particularly attractive with its unique capability of guiding the training with physical laws written in the forms of partial differential equations. Thermomechanical simulation for additive manufacturing (AM), a multi-scale, multi-physics problem could potentially benefit from the use of PINN, as demonstrated in some successful attempts in the literature. In this work, PINN is applied to different metal AM processes and several challenges that limit the robustness of PINN are observed. This paper aims to provide a summary of the observations and a preliminary attempt to account for such observations in order to pave the path for future work that aims to unleash the full promise of PINN in AM-related applications.

I. Introduction

Additive manufacturing (AM) has been envisaged as a key pillar of Industry 4.0 and identified as the enabling means for advanced design optimisation techniques (e.g. topology optimisation (TO)) with its tremendous design freedom offered by the additive nature compared to the conventional manufacturing methods [1]. However, the additive nature also imposes its own unique set of challenges for manufacturability – overhang angle constraints, lack of fusion, thermomechanically-induced defects (e.g. keyholing, spattering etc.), and failures induced by the deviation from original design during manufacturing (e.g. thermal distortion, blade collision etc.) [2]. Some challenges are universal for different processes while others are more pertinent to one process than the other.

The thermally-induced defects and interactions are particularly prominent in metal AM (MAM) due to the extremely rapid temperature gradients and the use of high energy sources (e.g. laser beams and electron beams). Print failures may occur if they are not adequately accounted for and/or when the process parameters are away from the trial-and-tested profiles [3]. Hence, modelling and predicting the phenomena associated with extreme thermal conditions have been an active area of research over the years to improve printability and the rate of 'first-time-right'.

The investigation of the thermally related behaviours in MAM is a complex subject – it is a multi-physics and multi-scale problem. Depending on the area of interest, the focus of the investigation ranges from mesoscale to part-scale and the type of analyses can be thermal, thermomechanical, thermo-metallugical, and/or thermo-fluid [1]. Depending on the scale of the

method, it could either be information rich but computationally expensive [4, 5] or fast to run but loses certain levels of detail or requires calibration [6, 7].

Despite the progress in computing power over the year, the current hardware is still unable to support part-level, multi-scale thermomechanical simulation. In the meantime, the rapid development of machine learning (ML) based methods has consequently made its way into the field of AM. Leveraging on its capability ranging from design generation to surrogate modelling, ML methods are utilised for inverse design, print simulation, defect prediction etc. [8–10].

Physics-informed Neural Network (PINN) was first proposed in 2019 [11]. Unlike typical ML models, PINN allows the training to be constrained by physical laws expressed in partial differential equations (PDEs). Hence, it is believed to be less of a black box which is a key concern over the use of ML for simulation. It has been demonstrated that PINN can be used without any auxiliary data to solve many common and even challenging PDE problems (e.g. Burger's equation, heat transfer, Navier-stokes equation etc.), or solve the inverse problem where the physical parameters can be approximated by learning from data and guided by the PDEs [12]. In the field of thermomechanical simulation for AM, there have been a few attempts in applying PINN as an alternative to numerical simulations. In [13], Liao et al. proposed a hybrid method that predicts the temperature evolution during the Direct Energy Deposition (DED) process. In [14], Zhu et al. employed PINN to approach the micro-scale thermal problem where the melt pool dimension is approximated.

The authors believe that PINN offers a unique advantage over the standard numerical methods with its mesh-less characteristics in solution and more importantly, the possibility of obtaining the derivatives of the output with regard to the inputs relatively cheaply for a trained model – it implies that optimisation of design parameters with regard to the output (e.g. factors affecting the printability) could be more achievable.

With the positive outlook of PINN in mind, this paper reports the major observations in applying PINN for temperature field prediction of MAM processes – the key challenges that limit the transferability of the current PINN-based method from one MAM process to another and some preliminary results on the attempts to resolve the issue.

The paper is divided into the following sections: Section II introduces the general methodology in applying PINN to solve the heat transfer during AM; Section III presents the preliminary results and discussions based on the results that highlight some challenges in the application of PINN; Section IV presents the future work proposed; and it is concluded with a compiled summary of the discussion.

II. Methodology
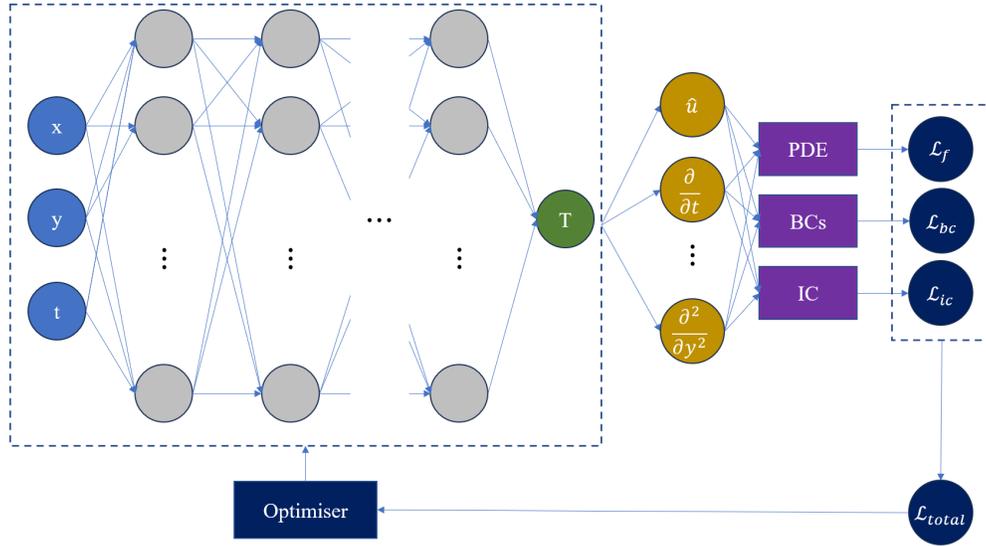
Fig. 1 Framework of PINN

The problem investigated in this work is an application of PINN in the forward problem. Figure 1 illustrates the framework of the setup. It takes advantage of the capability of auto-differentiation [15] of neural networks to obtain the respective terms in the transient heat equation and boundary condition equations as detailed in Section II.A. Due to the limited time, the investigation presented in this manuscript is based on a 2D domain. However, work in 3D has been carried out and will either be presented at the conference or in the journal paper. The PINN-based method for temperature history is applied to two common MAM processes – Direct Energy Deposition (DED) and Laser Power Bed Fusion (L-PBF).

A. Governing Equations of Heat Transfer

The basis of this work is transient heat conduction which is governed by the following equation:

$$\rho C_p \frac{\partial T}{\partial t} + \nabla \cdot \mathbf{q} = 0 \tag{1}$$

where ρ is the density of the material, C_p is the specific heat capacity, T is temperature and $\mathbf{q}$ is heat flux.

Applying the Fourier's Law:
$$\mathbf{q} = -k(T)\nabla T$$

Equation 1 in the 2D case can be expended to:

$$\rho C_p \frac{\partial T}{\partial t} - \frac{\partial k}{\partial T}\left(\left(\frac{\partial T}{\partial x}\right)^2 + \left(\frac{\partial T}{\partial y}\right)^2\right) - k(T)\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right) = 0 \tag{2}$$

where k is the heat conductivity of the material. In many cases, it is assumed that k is constant with temperature and Equation 2 can be further simplified to:

$$\rho C_p \frac{\partial T}{\partial t} - k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) = 0 \tag{3}$$

In this work, both the constant heat conductivity and temperature-varying heat conductivity are implemented to compare and highlight the effect of varying heat conductivity during the large temperature change during AM process.

For a unique solution, boundary conditions (BC) and initial conditions (IC) have to be specified.

The boundaries of the domain are of two types: standard boundary and boundary where the heat source is applied. For the former, depending on the process, either Neumann BC (Equation 4) or Dirichlet BC (Equation 4) are applied.

Neumann BC – the total heat flux into the system is equal to the sum of the external heat flux at the boundary:

$$\mathbf{q} \cdot \mathbf{n} = \sum q_{ext} \tag{4}$$

where

$$\sum q_{ext} = q_{conv} + q_{rad} = h(T - T_{amb}) + \sigma \varepsilon (T^4 - T_{amb}^4)$$

Dirichlet BC – the temperature at the boundary is equal to the reference temperature T_0:

$$T(\mathbf{x}) = T_0, \quad \mathbf{x} \in \partial \Omega \tag{5}$$

A commonly observed method to apply the heat source is by introducing the heat flux as a Neumann BC similar to that in Equation 4 with a change in the external heat flux term:

$$\sum q_{ext} = q_{conv} + q_{rad} - q_{laser}$$

where q_{laser} can be modelled with different level of complexity. In this work, the Gaussian model is employed [16] as Equation 6:

$$q_{laser} = \frac{2\eta P}{\pi r_{beam}^2} exp \left(\frac{-2d^2}{r_{beam}^2} \right) \tag{6}$$

where P is the laser power, η is the laser absorptivity, r_{beam} is the radius of the laser spot, and d is the Euclidean distance between a point and the centre of the laser spot.

In addition to the heat flux approach, the authors also propose introducing the heat source as a Dirichlet pseudo-boundary condition where a pre-defined field (melt pool (mp)) of temperature is applied to the domain. It bypasses a key issue in the L-PBF case which will be discussed further in Section III.A.

$$T(\mathbf{x}) = T_{mp}, \quad \mathbf{x} \in \Omega_{mp} \tag{7}$$

The Dirichlet initial condition is applied, assuming the domain is at the ambient temperature when the simulation starts:

$$T|_{t=0} = T_{amb} \tag{8}$$

B. Physics Informed Neural Network Construction

The PINN used in this work is based on a fully connected neural network which is one of the most common types of neural network architecture in PINN applications. The solution from PINN can be generalised as:

$$\hat{u}(\mathbf{x}, t) \approx T(\mathbf{x}, t) \tag{9}$$

The governing equations for the PDE, Neumann BC, Dirichlet BC, and IC discussed in the previous section can be written in the following general forms, respectively:

$$\mathcal{F}(u, \mathbf{x}, t) = u_t + \mathcal{N}[u] = 0 \tag{10}$$

$$\mathcal{B}_{neum}(u, \mathbf{x}, t, \lambda) = \mathcal{N}[u] + \lambda = 0 \tag{11}$$

$$\mathcal{B}_{dir}(u, \mathbf{x}, t) = u(\mathbf{x}, t) - B(\mathbf{x}, t) \tag{12}$$

$$\mathcal{I}(u, \mathbf{x}) = u(\mathbf{x}, 0) - I(\mathbf{x}) \tag{13}$$

where $\mathcal{N}$ indicates a general differential operator, λ is a general representation of known physical quantities (e.g. sum of all external heat fluxes), B is the Dirichlet BC, and I is the initial condition.

The individual loss terms for the training of the PINN can subsequently be written as:

$$\mathcal{L}_f = \frac{1}{N_f} \sum_{i=1}^{N_f} \left| \mathcal{F}\left(\hat{u}(\mathbf{x}_f^i, t_f^i), \mathbf{x}_f^i, t_f^i\right) \right|^2 \tag{14}$$

$$\mathcal{L}_{bc} = \frac{1}{N_{bc_{neum}}} \sum_{i=1}^{N_{bc_{neum}}} \left| \mathcal{B}_{neum}\left(\hat{u}(\mathbf{x}_{bc_{neum}}^i, t_{bc_{neum}}^i), \mathbf{x}_{bc_{neum}}^i, t_{bc_{neum}}^i, \lambda_{bc_{neum}}^i\right) \right|^2 +$$

$$\frac{1}{N_{bc_{dir}}} \sum_{i=1}^{N_{bc_{dir}}} \left| \mathcal{B}_{dir}\left(\hat{u}(\mathbf{x}_{bc_{dir}}^i, t_{bc_{dir}}^i), \mathbf{x}_{bc_{dir}}^i, t_{bc_{dir}}^i\right) \right|^2 \tag{15}$$

$$\mathcal{L}_{ic} = \frac{1}{N_{ic}} \sum_{i=1}^{N_{ic}} \left| \mathcal{I}\left(\hat{u}(\mathbf{x}_{ic}^i, 0), \mathbf{x}_{ic}^i\right) \right|^2 \tag{16}$$

where N_f, $N_{bc_{neum}}$, $N_{bc_{dir}}$, and N_{ic} are the number of collocation points sampled to compute the PDE, BC, and IC losses.

The total loss used for the training of the PINN is obtained by assembling the individual loss terms through a weighted sum:

$$\mathcal{L}_{total} = w_f * \mathcal{L}_f + w_{bc} * \mathcal{L}_{bc} + w_{ic} * \mathcal{L}_{ic} \tag{17}$$

It should be noted that should auxiliary data be included, an additional loss term for data can be added similarly. It is omitted in this report as no auxiliary data is used.

III. Preliminary Results and Discussion

Unless otherwise specified, the following configurations are applied for the DED and L-PBF cases, respectively.

Table 1   Basic parameters settings

		DED	L-PBF
Heat Flux Source	r_{laser} **[mm]**	1.5	0.08
	v_{laser} **[mm/s]**	10	1000
	P [W]	500	200
	Output Transform	$3000 * SoftPlus() + T_{amb}$	
Melt Pool Source	l_{mp} **[mm]**	/	0.522
	d_{mp} **[mm]**	/	0.076
	v_{mp} **[mm/s]**	/	1000
	T_{mp} **[K]**	/	3000
	Output Transform	/	Temperature Normalisation

A. Effect of scale

(a) Snapshot of output (t=1.33s)
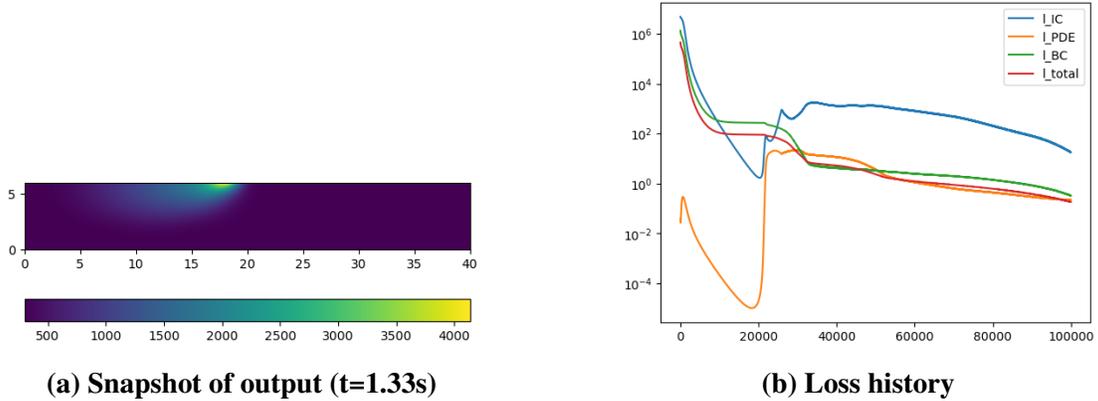
(b) Loss history

Fig. 2   DED, Heat Flux Source, with Output Transform
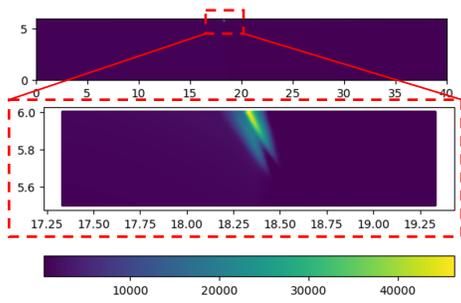
(a) Snapshot of output (t=1.33s)
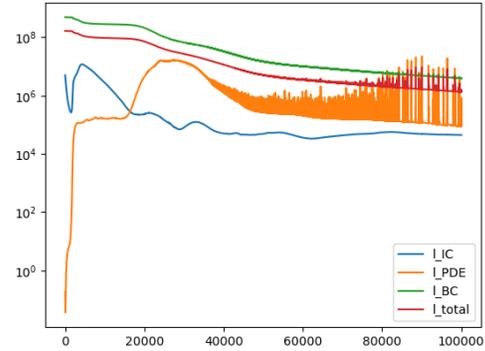
(b) Loss history

Fig. 3   L-PBF, Heat Flux Source ($r_{laser} = 50\mu m$, $v_{laser} = 10mm/s$), with Output Transform
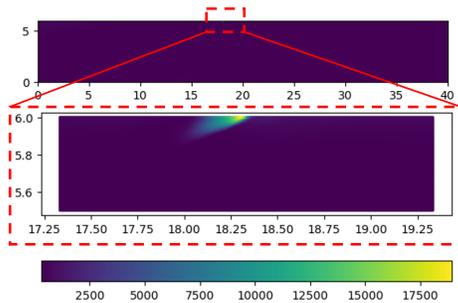
(a) Snapshot of output (t=0.0133s)
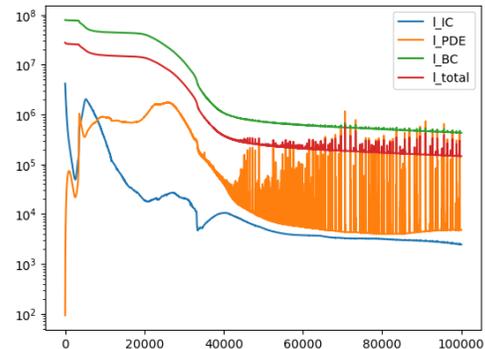
(b) Loss history
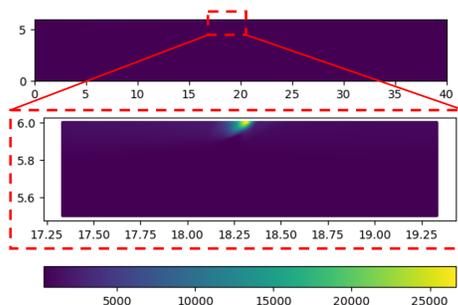
(c) Snapshot of output (t=0.0133s)
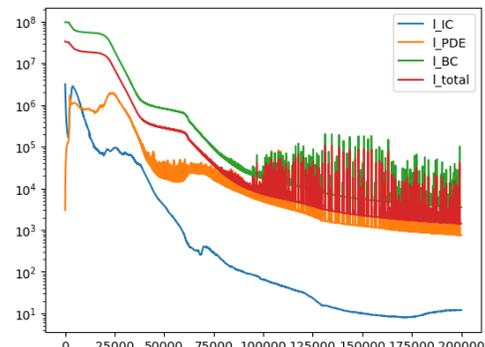
(d) Loss history

Fig. 4   L-PBF, Heat Flux Source, with Output Transform, 40×6mm Domain

(a) Snapshot of output (t=0.00267s)

(b) Loss history

(c) Snapshot of output (t=0.00267s)

(d) Loss history
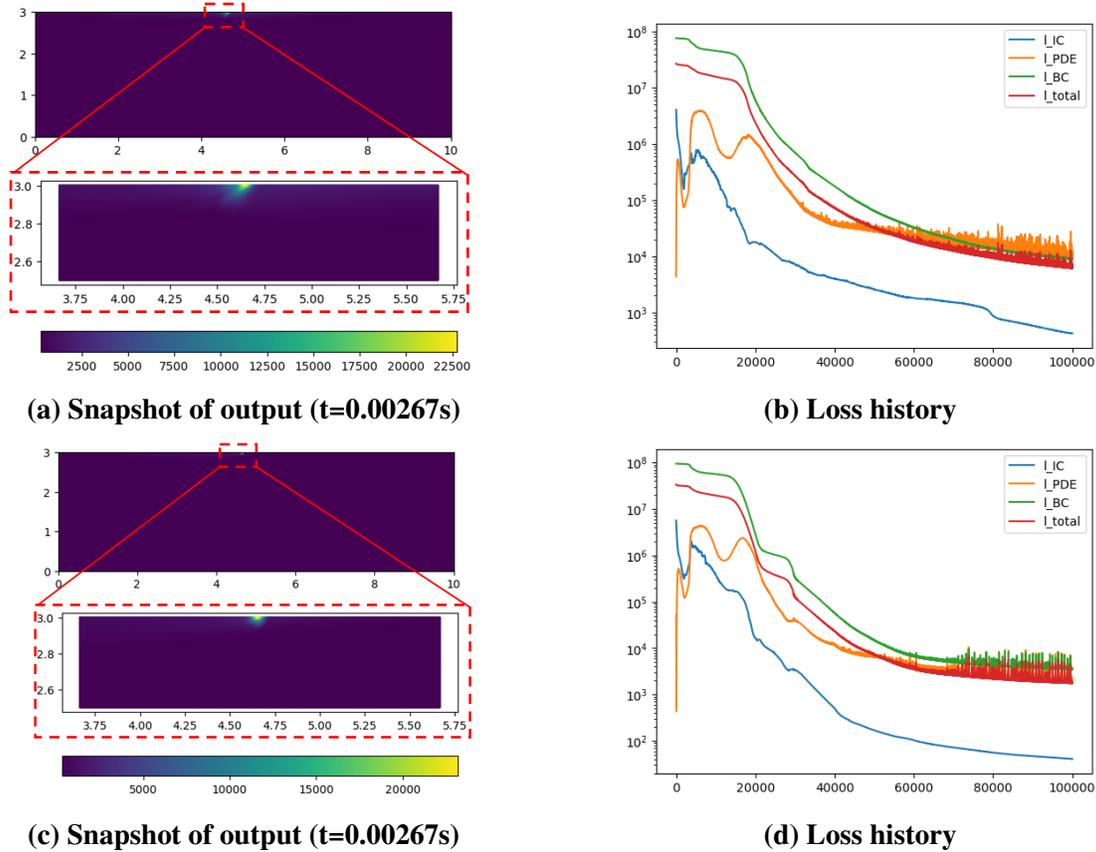
Fig. 5   L-PBF, Heat Flux Source, with Output Transform, 10×3mm Domain

One major observation from the experiment is the sensitivity of training to scale (both spatial and temporal). While the model can be trained relatively well for the DED process, as exemplified in both [13] and the outputs shown in Figure 2, the same model will not produce correct results when the process is changed to L-PBF, despite that the few changes between the two processes are just the laser spot size, laser travel speed, laser power from modelling's perspective, and the boundary conditions for the sides.

By comparing the results in Figure 2 and 3, it can be concluded that the most important factor that affects the training is the length scale of the laser spot size – when the laser spot size becomes too small, the heat diffusion can no longer be captured and the peak temperature becomes unrealistically high. Such observation is analogous to the instability in numerical schemes where the solution does not converge when the spatial and temporal resolutions are too coarse to capture the features in the simulation. However, refining the collocation points (analogues to mesh refinement in numerical methods) does not necessarily always lead to convergence in the context of PINN. Besides the common problem as that for numerical methods where the number of collocation points (or mesh resolution) cannot be refined indefinitely, PINN will be affected by the distribution of the collocation points. It implies that merely increasing the number of collocation points will not always improve the results as the spatial variation of the error values will be compressed and lost during the L2-loss computation which is subsequently used to guide the training of the NN-approximated

function. Hence, strategically distributing the collocation points at the more critical areas (i.e. where the heat source is) becomes another critical tuning process. The comparison between Figure 4 and 5 indicates that simply reducing the domain size or increasing the number of collocation points does not necessarily lead to improvement in results.
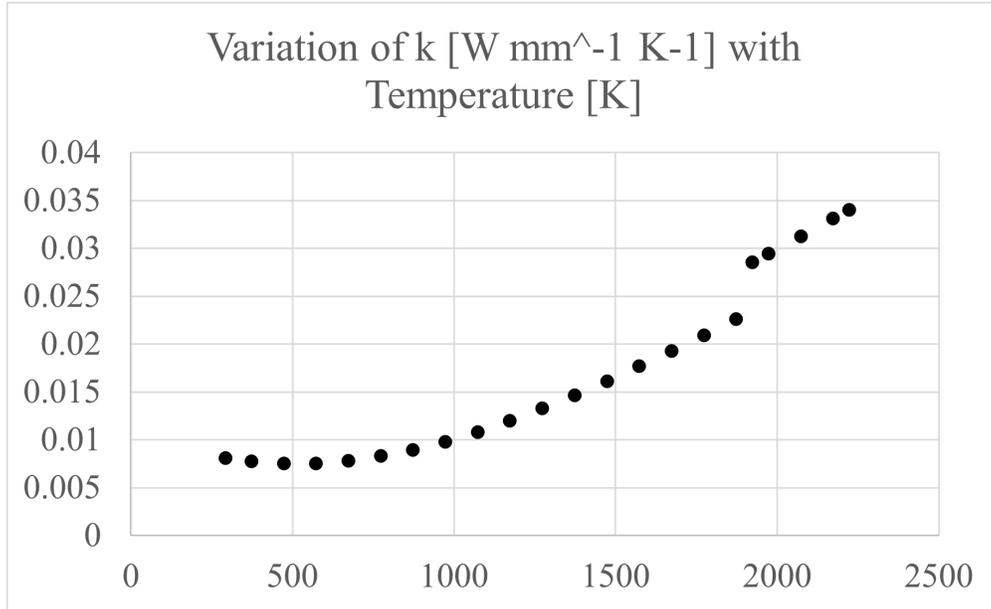
Fig. 6 Variation of heat conductivity with temperature [17]
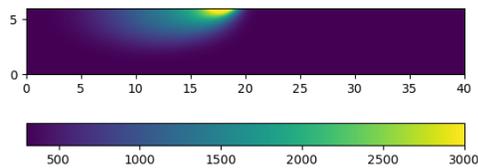
(a) Constant k - Snapshot of output (t=1.33s)
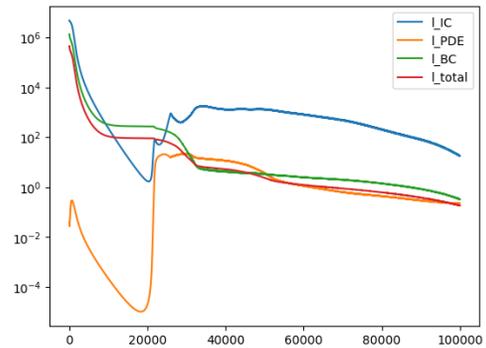
(b) Constant k - Loss history
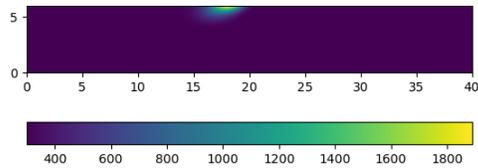
(c) Varying k - Snapshot of output (t=1.33s)
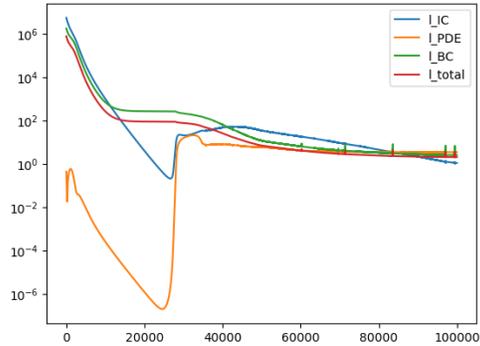
(d) Varying k - Loss history

Fig. 7   DED, Heat Flux Source, with Output Transform

In addition to tuning the collocation points, a more fundamental reason for the failure of PINN in the cases of L-PBF could potentially be the inaccurate physics that is embedded in the training. Equation 1 describes the heat transfer through conduction. While it is generally true when the laser (or other beams) spot size is large (i.e. less concentrated heat and hence a melt pool size that is comparable to the laser spot size), it is no longer accurate when the laser spot size is much smaller than the melt pool size – convection of the molten metal in the melt pool has a more prominent role and has to be accounted for. The characteristics of PINN being constrained by the physical PDE(s) in the loss computation imply that the solution from PINN will not be accurate when the PDE(s) constraining the training is/are incorrect. It is especially the case in forward problems where no auxiliary experimental data is given. One potential solution besides explicitly modelling the convection in the melt pool is to obtain an 'effective heat conductivity' that accounts for the heat transfer in the melt pool [18]. However, the determination of the temperature-dependent, process-dependent parameter requires extensive experiments and calibration which is beyond the scope of this work. Instead, the values from ANSYS material database (illustrated in Figure 6) are used to demonstrate the effect of accounting for the additional heat transfer. As shown in Figure 7, the peak temperature is reduced and the result is improved. It should, however, be acknowledged that such a method will not improve the robustness of using PINN to solve for the temperature history with the heat flux as the heat input if frequent calibration is required.

B. Output Transform

Another key observation is the critical role of output transform. In [13], it is mentioned that '... the Softplus function is used to ensure a positive output ...'. Upon closer inspection of the code available on GitHub, a coefficient of 3000 is applied to the output transform. It is hypothesised that the coefficient is put in place to balance the temperature-gradient-based losses and temperature-based losses to similar orders of magnitude as it is observed that a large difference in the order of magnitude between losses could prevent training as detailed in Section III.C.

(a) Snapshot of output (t=1.33s)
$3000 * SoftPlus() + T_{amb}$

(b) Loss history
$3000 * SoftPlus() + T_{amb}$

(c) Snapshot of output (t=1.33s)
$SoftPlus() + T_{amb}$

(d) Loss history
$SoftPlus() + T_{amb}$

(e) Snapshot of output (t=1.33s)
No Output Transform

(f) Loss history
No Output Transform
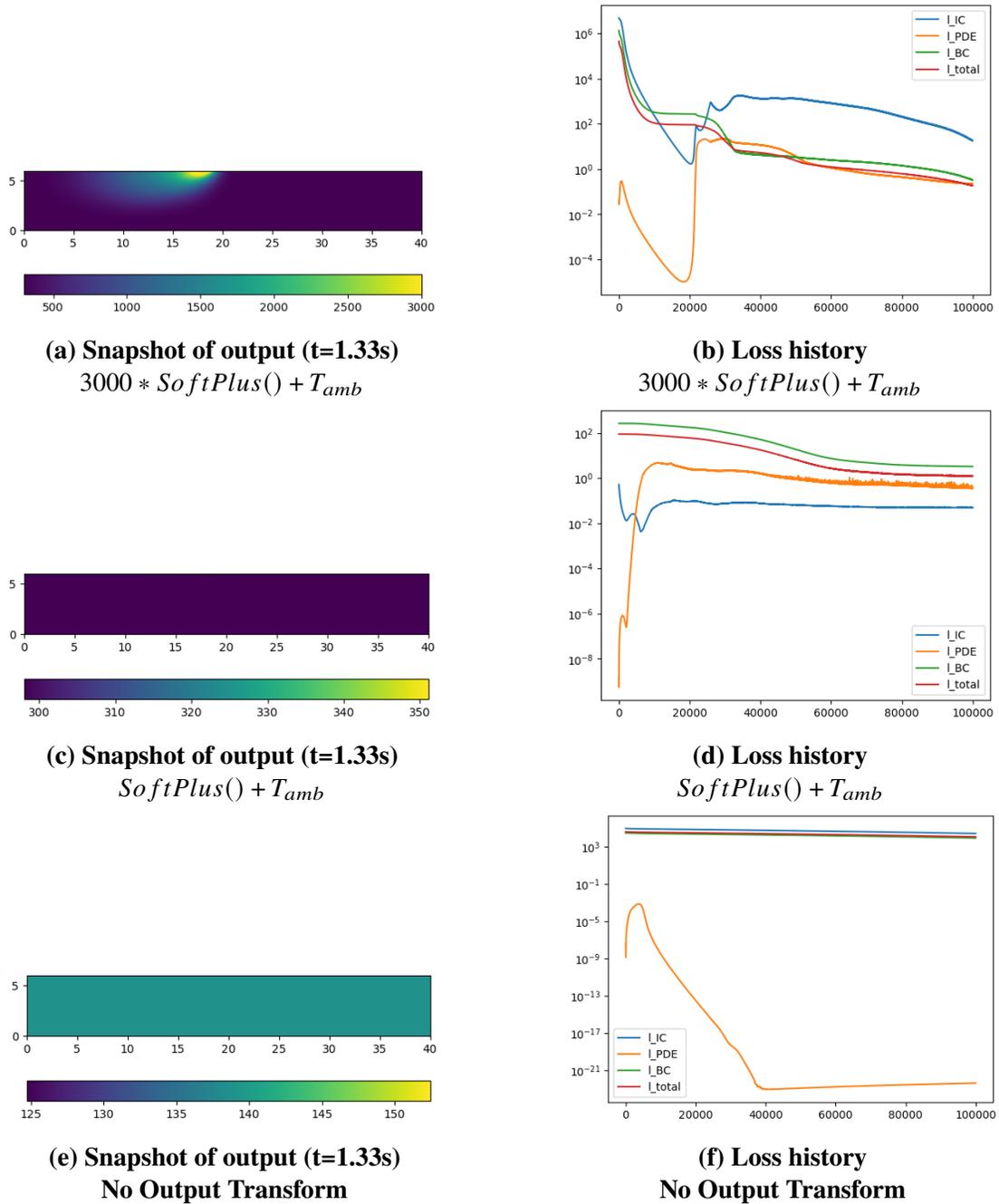
Fig. 8　DED, Heat Flux Source

Figure 8 illustrate the effect of the *SoftPlus()* based output transform. It is evident that the transform or more importantly, the coefficient of the transform plays a critical role in ensuring the successful convergence of the result. However, the authors were unable to conclude a systematic method of determining the coefficient for different cases. It implies that the value of the coefficient could become an additional hyperparameter which is less ideal. Figure 9 illustrates the effect of setting the coefficient based on the difference between the temperature-gradient-based and the

temperature-based errors. The effect is not significant and more investigation is required.

(a) Snapshot of output (t=0.0133s)
$3000 * SoftPlus() + T_{amb}$

(b) Loss history
$3000 * SoftPlus() + T_{amb}$

(c) Snapshot of output (t=0.0133s)
$30000 * SoftPlus() + T_{amb}$

(d) Loss history
$30000 * SoftPlus() + T_{amb}$
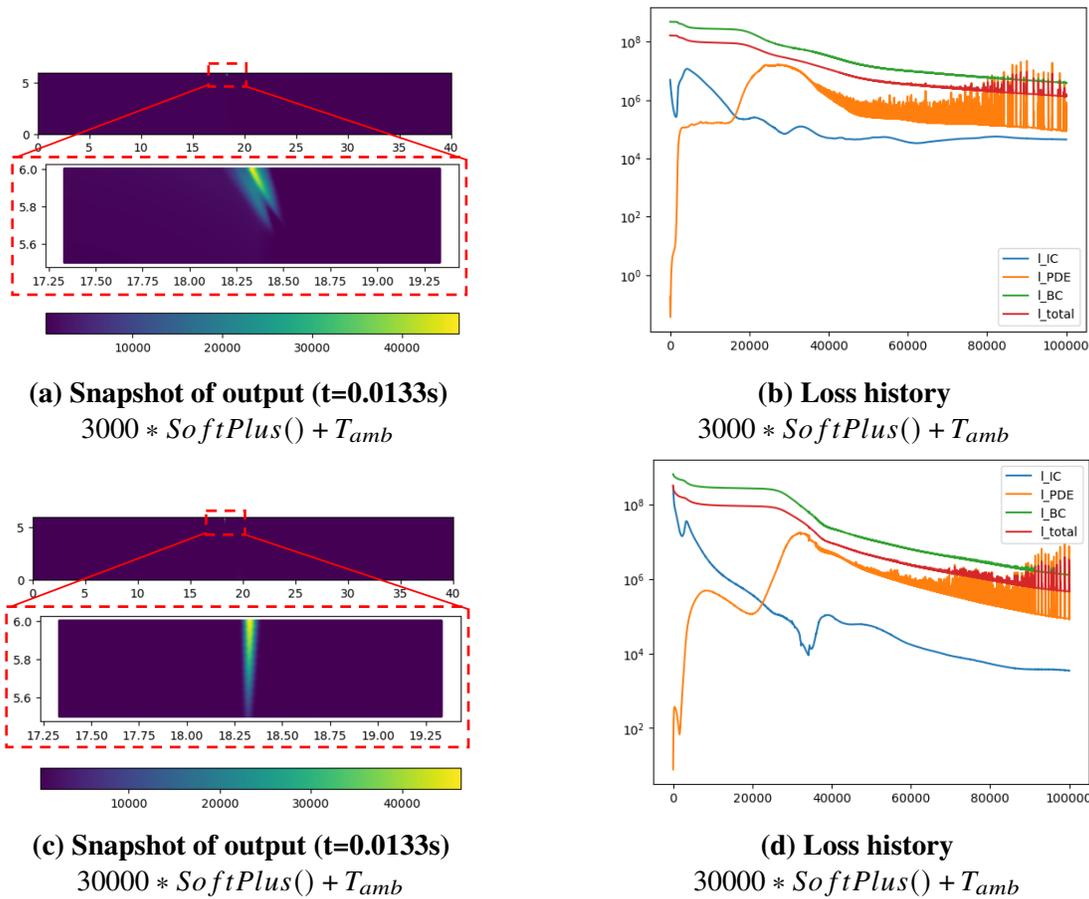
Fig. 9   L-PBF, Heat Flux Source

C. Indicator of Successful Training

At the core of PINN training, it is a multi-objective optimisation of the various loss terms that are governed either by the physical description (i.e. in the form of PDEs) or auxiliary data. However, depending on the setup of the problem and the validity of assumptions, the losses could be computed from different sets of collocation points. It implies that the different physical phenomena (e.g. heat flux from the boundary and heat transfer within the body) could be entirely uncoupled and the model is trained to fit them discretely in the extreme cases. In these cases, the model essentially converges to a solution that is analogous to a trivial solution.

Hence, an important indicator for the potential successful training is that the losses converge to a similar order of magnitude before decreasing together, as illustrated by the loss history plots in the previous sections. Hence, there are at least two factors dictating the harmonious convergence of losses – the distribution of the collocation points as well as the magnitude of the different losses. The former is intuitive as the spatiotemporally close collocation points imply better overlapping of the prescribed physical equations to be satisfied. It should be noted that collocation points do

not have to necessarily overlap as long as the gaps between domains are not too big, empirically speaking. When there is a large discrepancy in the magnitude of losses, there is a limit beyond which the lower loss(es) (e.g. the PDE loss) will not rise sufficiently to the similar order of magnitude as the higher loss(es) (e.g. the BC loss). The model will then converge to a local minimum where the training for different loss terms becomes uncoupled. The authors attribute the observation to the error being saturated when there is a large discrepancy, preventing further training.

D. Soft vs. Hard Constraint

(a) Snapshot of output (t=0.0133s)
soft constraint

(b) Loss history
soft constraint

(c) Snapshot of output (t=0.0133s)
hard constraint

(d) Loss history
hard constraint
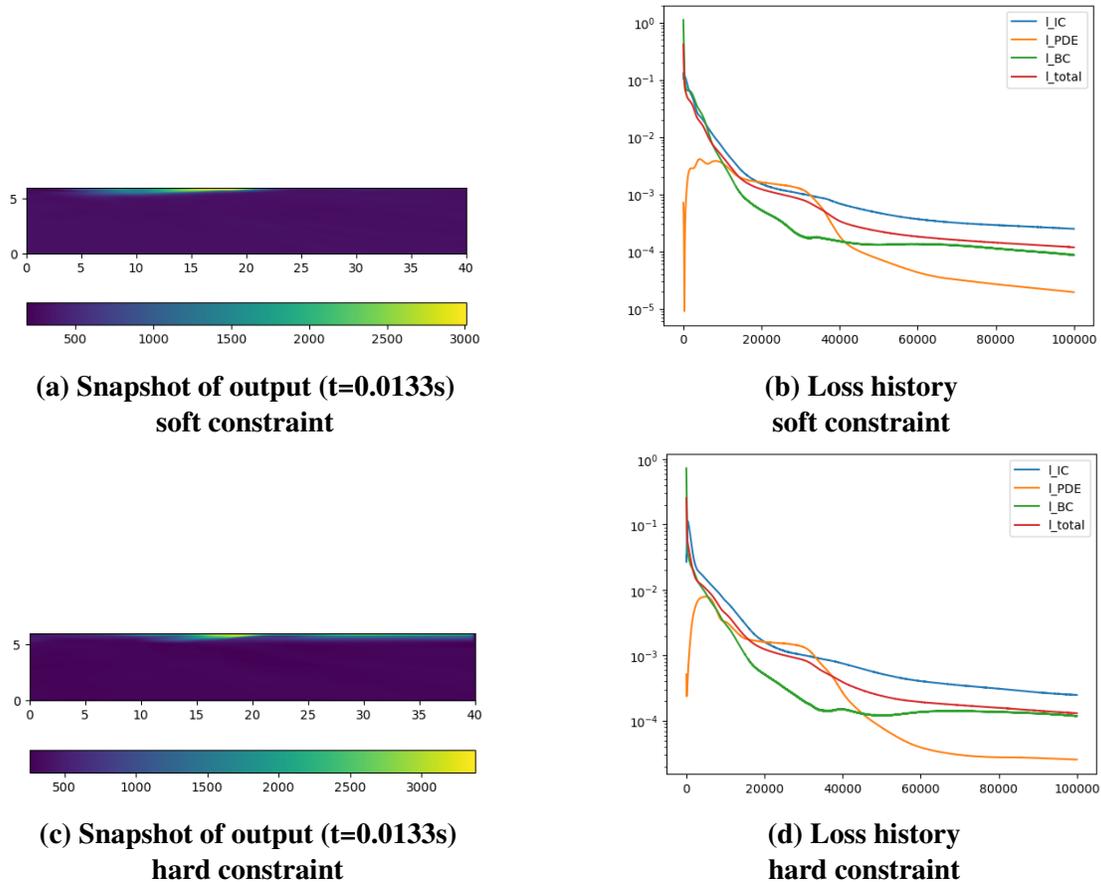
Fig. 10   L-PBF, Melt Pool Source

Applying hard constraints is another method to enforce the boundary conditions in PINN. As the name suggests, instead of using loss terms for the BC to guide the training which will not guarantee the BC being fully met, hard constraints can be introduced by modifying the model architecture (commonly done through input transform or output transform) such that the boundary conditions are always met – usually through shape functions. It has been demonstrated to reduce the training effort in many cases [19]. However, as shown in Figure 10, imposing hard constraints does not necessarily lead to improved training or outputs. More investigation is hence needed to determine the cause.

IV. Conclusion

The discussion from the previous sections highlights that training a PINN to solve for the temperature history during MAM is not a trivial task, especially for a training procedure that is robust enough for different MAM processes. Besides the common ordeal of hyperparameter tuning that is present in almost all ML-based applications, the self-supervising nature of solving a forward problem using PINN, and the wide spectrum of loss terms imposes additional challenges. The actions that can be taken to improve the convergence can be summaries into three main classes. It should also be highlighted that actions taken in one class could augment or negate the effect of those in another.

1. Collocation Point: As discussed in Section III.A and III.C, the density and distribution of collocation points can both have a significant impact on the outcome of the training. It is also widely acknowledged in the literature [11, 20]. However, the determination of the exact number of collocation points is essentially trial and error. A more systematic approach (e.g. adaptive collocation refinement) could be employed to improve the effectiveness and efficiency. It should also be acknowledged that the distribution of collocation points can have an impact on the weight of the different losses which is a key element in hyperparameter tuning.

2. Loss Hamonisation: The phenomenon described in Section III.C highlights a common sight in PINN where a spectrum of loss magnitude could be present. Therefore, it is ideal to reduce the difference in magnitude between the losses. As demonstrated, the application of output transform such as that in [13] could be an effective solution but might be only effective for the specific problem. There have been some theoretical attempts made to account for the causes of the observation and consequently propose strategies to fix the issue [21, 22]. More investigations on the effectiveness of such strategies will be conducted.

3. Hyperparameter Tuning: Similar to any other ML application, hyperparameter tuning plays an important role in improving the training process and even affecting the outcome of the training. The multi-objective characteristics of PINN imply that the weights for the individual loss terms can act as hyperparameters that directly affect the training process as indicated in Equation 17. The tuning of the weights should be strategic since it will affect the effect of changing collocations and controlling the losses' magnitude.

Hence, as a way forward to reap the perceived benefit of PINN-based solutions, the focus of the upcoming work will be more thorough investigations on strategies to disentangle the possible means to improve the convergence and robustness of the PINN models for different MAM processes, as well as methodologies to scale up the model for more complex geometries and scan patterns.

References

[1] Bayat, M., Dong, W., Thorborg, J., To, A. C., and Hattel, J. H., "A review of multi-scale and multi-physics simulations of metal additive manufacturing processes with focus on modeling strategies," *Additive Manufacturing*, Vol. 47, 2021, p. 102278. https://doi.org/10.1016/J.ADDMA.2021.102278.

[2] Panesar, A., Carollo, V., and Jamal, M., *Simulation-Driven Design and the Role of Optimization in Design for Additive Manufacturing*, ASM International, 2023. https://doi.org/10.31399/asm.hb.v24A.a0006950.

[3] Chahal, V., and Taylor, R. M., "A review of geometric sensitivities in laser metal 3D printing," *Virtual and Physical Prototyping*, Vol. 15, No. 2, 2020, pp. 227–241. https://doi.org/10.1080/17452759.2019.1709255, URL https://doi.org/10.1080/17452759.2019.1709255.

[4] Bayat, M., Mohanty, S., and Hattel, J. H., "Multiphysics modelling of lack-of-fusion voids formation and evolution in IN718 made by multi-track/multi-layer L-PBF," *International Journal of Heat and Mass Transfer*, Vol. 139, 2019, pp. 95–114. https://doi.org/10.1016/j.ijheatmasstransfer.2019.05.003.

[5] Denlinger, E. R., Gouge, M., Irwin, J., and Michaleris, P., "Thermomechanical model development and in situ experimental validation of the Laser Powder-Bed Fusion process," *Additive Manufacturing*, Vol. 16, 2017, pp. 73–80. https://doi.org/10.1016/j.addma.2017.05.001.

[6] Parry, L., Ashcroft, I., and Wildman, R., "Understanding the effect of laser scan strategy on residual stress in selective laser melting through thermo-mechanical simulation," *Additive Manufacturing*, Vol. 12, 2016, pp. 1–15. https://doi.org/10.1016/j.addma.2016.05.014.

[7] Dong, W., Liang, X., Chen, Q., Hinnebusch, S., Zhou, Z., and To, A. C., "A new procedure for implementing the modified inherent strain method with improved accuracy in predicting both residual stress and deformation for laser powder bed fusion," *Additive Manufacturing*, Vol. 47, 2021, p. 102345. https://doi.org/10.1016/J.ADDMA.2021.102345.

[8] Wang, J., and Panesar, A., "Machine learning based lattice generation method derived from topology optimisation," *Additive Manufacturing*, Vol. 60, 2022, p. 103238. https://doi.org/10.1016/j.addma.2022.103238.

[9] DebRoy, T., Mukherjee, T., Wei, H. L., Elmer, J. W., and Milewski, J. O., "Metallurgy, mechanistic models and machine learning in metal printing," *Nature Reviews Materials*, Vol. 6, 2020, pp. 48–68. https://doi.org/10.1038/s41578-020-00236-1.

[10] Mozaffar, M., Liao, S., Xie, X., Saha, S., Park, C., Cao, J., Liu, W. K., and Gan, Z., "Mechanistic artificial intelligence (mechanistic-AI) for modeling, design, and control of advanced manufacturing processes: Current state and perspectives," *Journal of Materials Processing Technology*, Vol. 302, 2022, p. 117485. https://doi.org/10.1016/J.JMATPROTEC.2021.117485.

[11] Raissi, M., Perdikaris, P., and Karniadakis, G. E., "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, Vol. 378, 2019, pp. 686–707. https://doi.org/10.1016/J.JCP.2018.10.045.

[12] Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., and Yang, L., "Physics-informed machine learning," *Nature Reviews Physics 2021 3:6*, Vol. 3, 2021, pp. 422–440. https://doi.org/10.1038/s42254-021-00314-5, URL https://www.nature.com/articles/s42254-021-00314-5.

[13] Liao, S., Xue, T., Jeong, J., Webster, S., Ehmann, K., and Cao, J., "Hybrid thermal modeling of additive manufacturing processes using physics-informed neural networks for temperature prediction and parameter identification," *Computational Mechanics*, 2023. https://doi.org/10.1007/s00466-022-02257-9.

[14] Zhu, Q., Liu, Z., and Yan, J., "Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks," *Computational Mechanics*, Vol. 67, 2021, pp. 619–635. https://doi.org/10.1007/s00466-020-01952-9.

[15] Baydin, A. G., Pearlmutter, B. A., Radul, A. A., and Siskind, J. M., "Automatic differentiation in machine learning: a survey," , 2018.

[16] Bayat, M., Mohanty, S., and Hattel, J. H., "A systematic investigation of the effects of process parameters on heat and fluid flow and metallurgical conditions during laser-based powder bed fusion of Ti6Al4V alloy," *International Journal of Heat and Mass Transfer*, Vol. 139, 2019, pp. 213–230. https://doi.org/10.1016/j.ijheatmasstransfer.2019.05.017.

[17] ANSYS, Inc., "Ansys GRANTA EduPack software," , 2022. URL https://https://www.ansys.com/.

[18] Shrestha, S., Cheng, B., and Chou, K., "An Investigation into Melt Pool Effective Thermal Conductivity for Thermal Modeling of Powder-Bed Electron Beam Additive Manufacturing," *Solid Freeform Fabrication 2016: Proceedings of the 27th Annual International Solid Freeform Fabrication Symposium*, 2016.

[19] Lu, L., Pestourie, R., Yao, W., Wang, Z., Verdugo, F., and Johnson, S. G., "Physics-informed neural networks with hard constraints for inverse design," , 2021.

[20] Cuomo, S., Cola, V. S. D., Giampaolo, F., Rozza, G., Raissi, M., and Piccialli, F., "Scientific Machine Learning Through Physics–Informed Neural Networks: Where we are and What's Next," *Journal of Scientific Computing*, Vol. 92, 2022, p. 88. https://doi.org/10.1007/s10915-022-01939-z.

[21] Wang, S., Yu, X., and Perdikaris, P., "When and why PINNs fail to train: A neural tangent kernel perspective," *Journal of Computational Physics*, Vol. 449, 2022, p. 110768. https://doi.org/10.1016/j.jcp.2021.110768.

[22] Zeng, Q., Kothari, Y., Bryngelson, S. H., and Schäfer, F., "Competitive Physics Informed Networks," , 2022.