CIRP Annals - Manufacturing Technology 00 (2023) 1–4

Contents lists available at ScienceDirect

CIRP Annals - Manufacturing Technology

journal homepage: https://www.editorialmanager.com/CIRP/default.aspx

Physics-informed deep learning of gas flow-melt pool multi-physical dynamics during powder bed fusion

Rahul Sharma[a,b], Maziar Raissi[c], Yuebin Guo (1)[a,b,*]

[a] Department of Mechanical and Aerospace Engineering, Rutgers University-New Brunswick, Piscataway, NJ 08854, USA
[b] New Jersey Advanced Manufacturing Institute, Rutgers University-New Brunswick, Piscataway, NJ 08854, USA
[c] Department of Applied Mathematics, University of Colorado, Boulder, CO 80309, USA

ARTICLE INFO

Article history:
Available online xxx

Keyword:
Selective laser melting
Dynamics
Physics-informed machine learning

ABSTRACT

The effect of inert gas on melt pool dynamics has been largely overlooked but is crucial for laser powder bed fusion (LPBF). Physics-based simulation models are computationally expensive while data-driven models lack transparency and need massive training data. This work presents a physics-informed deep learning (PIDL) model to accurately predict the temperature and velocity fields in the melting domain using only a small training data. The PIDL model can also learn unknown model constants (e.g., Reynolds number and Peclet number) of the governing equations. Furthermore, the robust PIDL algorithm converges very fast by enforcing physics via soft penalty constraints.

1. Introduction

The importance of inert gas flow (e.g., argon, nitrogen, or helium) is evident in laser powder bed fusion (LPBF) [1]. The recent results have revealed its significant influences on LPBF processes and consistent quality attributes, e.g., surface roughness, microstructure, porosity, and lack of fusion of the printed components [2–4]. The gas flow across the laser-melt pool interaction not only provides an inert atmosphere to prevent oxidation during printing but also transports microscale process by-products (e.g., plume, condensate, and spatters) away from the laser path to reduce beam scattering and attenuation (Fig. 1). While the increased gas flow speed will blow away the plume, the speed needs to be kept below a threshold to avoid particle pickup [5]. Furthermore, the variations in gas flow uniformity may impact the melt pool [6] and properties of the components [2].

Several computational fluid dynamics (CFD) models have been developed to simulate the velocity field of gas flow, improve its distribution uniformity in the build chamber [2], and evaluate the influence on soot and spatter particles [7]. Discrete phase models have also been developed to investigate spatter transport in a build chamber under Argon gas flow [8]. These computational models may help to understand the complex heat transfer phenomena between the gas-metal pool interface, but require the calibration of model parameters to account for the varying process conditions, which are computationally expensive. The high pressure, velocity, and surface temperature gradients coupled with the shear-driven action of gas flow drive Marangoni flows in the melt pool, which leads to process instability. These CFD models may simulate the velocity field of gas flow but the effect of gas flow on the melt pool dynamics is not analyzed well.

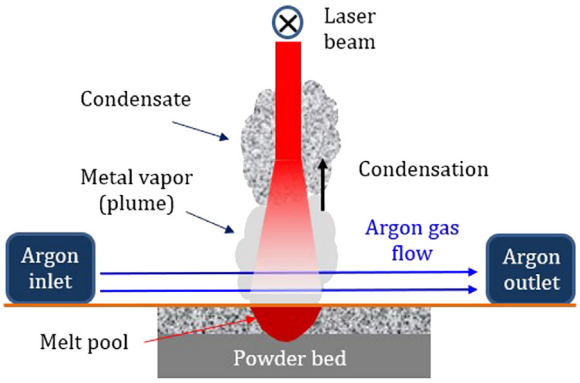

Fig. 1. Schematic of Argon (Ar) gas flow and process by-products in LPBF.

On the other hand, machine learning (ML) has the potential to handle the high dimensionality and big volume of online melt pool data (e.g., temperature) for process diagnosis, prognosis, and decision-making. However, the current pure data-driven ML algorithms suffer from the "black-box" nature – lacking process physics and explainability, inherently computation-intensive and storage-intensive (e.g., time-consuming), and the need of a large amount of training data to achieve an accurate prediction [9].

Initial efforts have been tried using a physics-informed neural network to solve LPBF problems [9–11]. However, a deep knowledge gap exists between ML and computational models for developing a data-efficient and explainable ML method to understand the complex multi-physical dynamics between the gas flow-melt pool in an LPBF process. To address this intractable problem, the three-fold objectives of this work are to: (1) develop a physics-informed deep learning (PIDL) framework by integrating deep learning and the physical laws underlying

* Corresponding author.
E-mail address: yuebin.guo@rutgers.edu (Y. Guo).

https://doi.org/10.1016/j.cirp.2023.04.005
0007-8506/

melt pool dynamics; (2) predict the temperature and velocity fields of the melt pool under the shear-driven influence of the gas flow; and (3) infer model constants of the governing equations of the melt pool with the known temperature and velocity fields.

2. PIDL methodology for multi-physical dynamics

2.1. Governing equations of gas shear-driven melt pool dynamics

This section presents the governing equations of gas shear-driven melt pool dynamics. As shown in Fig. 1, the Ar flows across the top surface of the melt pool. A rectangular domain of the melt pool is approximated for simplicity in this study. The top boundary is considered as the Ar moving with an assumed constant speed. To generalize the PIDL method to cover a broad range of LPBF conditions and powder materials, the governing partial differential equations (PDEs) are normalized using the maximum temperature and maximum velocity of the Argon which makes these equations independent of the maximum values of the involved parameters such as temperature, velocity, and pressure. Eqs. (1–4) represent energy conservation, momentum conservation, and mass conservation, respectively.

$$\frac{\partial T^*}{\partial t^*} + u^* \frac{\partial T^*}{\partial x^*} + v^* \frac{\partial T^*}{\partial y^*} - \frac{1}{Pe} \nabla^2 T^* = 0 \tag{1}$$

$$\frac{\partial u^*}{\partial t^*} + u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} + \frac{\partial P^*}{\partial x^*} - \frac{1}{Re} \nabla^2 u^* = 0 \tag{2}$$

$$\frac{\partial v^*}{\partial t^*} + u^* \frac{\partial v^*}{\partial x^*} + v^* \frac{\partial v^*}{\partial y^*} + \frac{\partial P^*}{\partial y^*} - \frac{1}{Re} \nabla^2 v^* = 0 \tag{3}$$

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \tag{4}$$

where T^*, u^*, v^*, and p^* represent the normalized melt pool temperature, x-component of velocity, y-component of velocity, and pressure, respectively. Pe and Re represent the Peclet number and Reynolds number and ∇^2 is the Laplace operator. To generate the datasets for training and testing the neural network (NN), COMSOL software is used to simulate the multiphysics problem. The melt pool flow is assumed to be laminar and values of Reynolds and Peclet numbers are 100 for this study [12]. The simulation has been performed from 0 to 4 s after which a steady-state fluid motion is achieved in the melting domain.

2.2. PIDL framework for multi-physical dynamic problems

In this study, a fully connected deep neural network (DNN) is used where every neuron in the hidden layer is connected to all the neurons of the previous and next layer (Fig. 2). In the hidden layer, the relation between the previous(n-1th) and current(n^{th}) layer output is given by:

$$z_n = \sigma_n(w_n^T z_{n-1} + b_n) \tag{5}$$

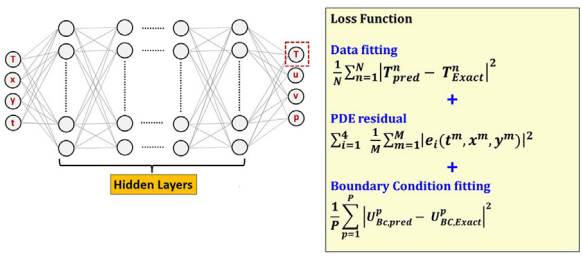

Fig. 2. PIDL architecture and loss function.

where $w_n$and b_n are the weights and biases for the current layer and $\sigma_n$is the swish activation function [10]. The input for the network is $\{t^n, x^n, y^n, T^n\}_{n=1}^N$ that corresponds to the value of passive scalar temperature at the spatial-temporal coordinates. In this study, temperatures at 10,000 scattered points per time step in the computation domain at 10 different time frames were used to train the model. The output of the network is $\{T^n, u^n, v^n\}_{n=1}^N$ which is the predicted temperatures and velocities at the same spatial-temporal coordinates. The novelty of the PIDL is manifested by its power of learning the velocity values without any training data except the used temperature data and boundary conditions. The temperature distribution in the melt pool is affected by the fluid flow which results in coupled energy and Navier Stokes equations. Due to this coupling of the governing equations, it is possible for PIDL to predict the velocity distribution in the melting domain by using the temperature data only. It is worth noting that many manufacturing phenomena like melt pool dynamics in LPBF are multiphysics problems where transport equations are coupled. In the case of LPBF, the surface temperatures of the melt pool can be collected using co-axial pyrometers or infrared cameras, one may predict the fields of other parameters such as velocity field in the melt pool without solving the non-linearly coupled Navier Stokes equations through the PIDL method. To calculate the derivatives of different parameters in the governing equations, an open-source software library Tensorflow has been used in this study. Tensorflow uses automatic differentiation instead of the Taylor series to compute the derivatives, which is fundamentally different from traditional numerical differentiation. This makes this PIDL method superior to conventional physics-based simulations.

The loss function (Fig. 2) has three terms – data fitting, residual of governing equations, and boundary conditions fitting. The data fitting loss function in terms of mean squared error (MSE) corresponds to the training data $\{t^n, x^n, y^n, T^n\}_{n=1}^N$ which utilizes the temperature data in the melting domain for training the network. The second loss term represents the penalty for four governing PDEs. The third loss term represents the MSE for the velocity boundary condition. The second and third loss functions make the PIDL a data-efficient model and help in the prediction of the velocity field without any additional training data. Compared with the conventional neural networks, after predicting the temperature and velocity values in each epoch, the PIDL model gives an extra penalty to the loss if Eqs. (1–4) are not satisfied by the predicted data. In this way, the loss function will converge very fast, and less training data is required to train the PIDL model when compared to the conventional machine learning models.

3. Forward learning of melt pool temperature and velocity

3.1. Ar flow shear-driven cavity

The effect of Ar flow on the melt pool dynamics is investigated to assess the performance of PIDL. Fig. 3 shows the comparison between the exact temperatures and velocities calculated from computational fluid dynamics (CFD) simulations with those predicted from the PIDL model. The “perfect” match between the exact data and predicted data shows the accuracy of the PIDL neural network to learn the temperature and velocity fields. It should be mentioned that only the scattered data of temperature is used for training and PIDL can predict the exact velocity field in the domain. This capability of the PIDL model is very useful for additive manufacturing and other industrial processes where physics-based simulations are very time-consuming and expensive due to the complex physics and coupled governing equations.

3.2. Effect of boundary conditions on PIDL model prediction

For certain multiphysics problems, PIDL may predict the velocity field accurately even without using the boundary conditions [13]. However, if boundary conditions for velocities are not enforced in this study, PIDL does not predict very well near the boundaries (Fig. 4) when compared with the predictions using boundary conditions. This is due to the fact that there is a steep velocity gradient at the top boundary due to which it is not converging to a single solution. Therefore, it would be very necessary to enforce boundary conditions in a PIDL model where steep gradients are present at the boundaries.

Please cite this article as: R. Sharma et al., Physics-informed deep learning of gas flow-melt pool multi-physical dynamics during powder bed fusion, CIRP Annals - Manufacturing Technology (2023), https://doi.org/10.1016/j.cirp.2023.04.005

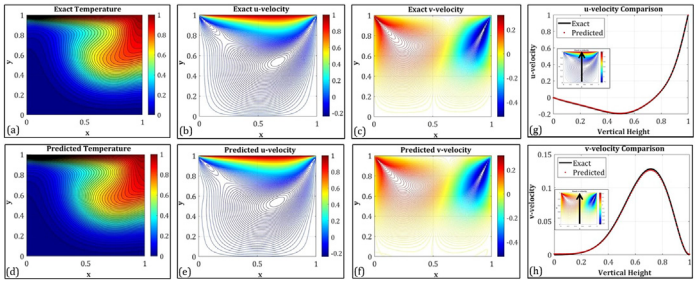

Fig. 3. Comparison of the exact values (a-c) and PIDL predictions (d-f) of T, u, and v contours @ t = 4 s and u and v profiles (g-h) along the center line. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

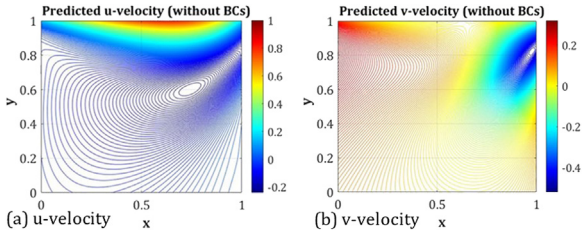

Fig. 4. Velocity predictions without enforcing BCs. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.3. Effect of training data on PIDL model predictions

To check the robustness of the PIDL algorithm, a systematic study has been performed with respect to the spatial-temporal resolution of the temperature where the different amounts of data at different amounts of time frames were fed to the model. Fig. 5 shows that algorithm breaks down if less than 900 data points and 6 time-frames are used. It suggests a threshold combination (e.g., [900,6] for this case) of training data points and time frames exist for an accurate prediction. Here the relative L_2 error is defined in terms of the predicted function $f(x_i)$ and exact function $g(x_i)$.

$$L_2 = \left(\frac{1}{N}\sum_{i=1}^{N}[f(x_i) - g(x_i)]^2\right) / \left(\frac{1}{N}\sum_{i=1}^{N}\left[g(x_i) - \sum_{i=1}^{N} g(x_i)\right]^2\right) \quad (6)$$

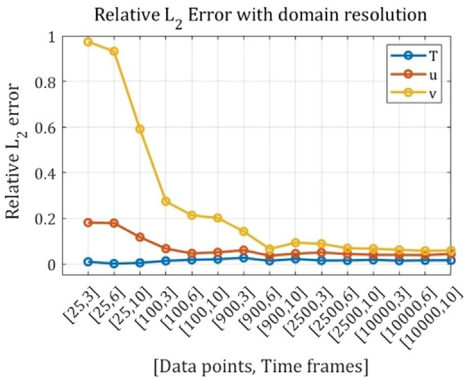

Fig. 5. Effect of spatial-temporal resolution on the relative L_2 error.

3.4. Model training efficiency and the effect of the NN architecture

Unlike conventional data-driven machine learning models with the inherent lengthy training time, the PIDL model converges very fast with a much less amount of data. Figs. 6 and 7 represents the data efficiency of the PIDL algorithm. This innovation is attributed to the soft penalty to the PIDL model for not satisfying the governing equations. The PIDL model in this study converges after only 6000 iterations that need approximately 16 min to complete. This is a dramatic improvement in training efficiency compared with the training time at the scale of hours/days for conventional ML. The loss function convergence helps in determining the optimized training time for the model.

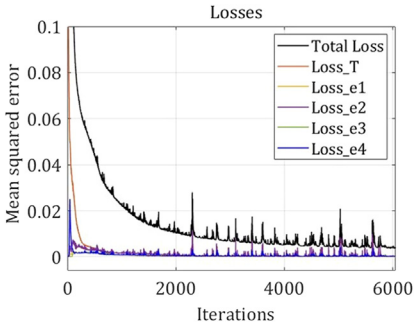

Fig. 6. Model training efficiency in terms of mean squared error (MSE). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

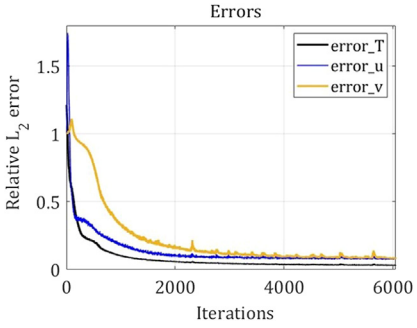

Fig. 7. Model training efficiency in terms of the relative L_2 error.

To assess the effect of a NN architecture on the prediction error, a systematic study was conducted with different combinations of hidden layers and neurons per layer (Fig. 8). It was found that 10 hidden layers with 200 neurons per layer are sufficient for the PIDL algorithm to converge. It is noted that other combinations of hidden layers and neurons may also lead to convergence.

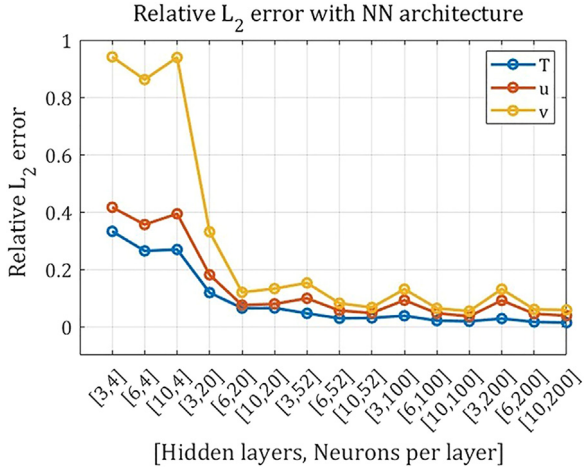

Fig. 8. Effect of the NN architecture on the relative L2 error.

Please cite this article as: R. Sharma et al., Physics-informed deep learning of gas flow-melt pool multi-physical dynamics during powder bed fusion, CIRP Annals - Manufacturing Technology (2023), https://doi.org/10.1016/j.cirp.2023.04.005

4. Inverse learning model constants of governing equations

In many practical problems, model constants of the governing equations are generally unknown or just roughly estimated values. In this case, the estimated values of Reynolds number (*Re*) and Peclet number (*Pe*) in Eqs. (1–3) were used for forward learning of melt pool temperature and velocity fields. However, accurate values of *Re* and *Pe* are very challenging to determine by either experimental or physics-based simulation methods considering the highly dynamic nature of the harsh melt pool. Multiphysics simulations of complex practical problems like LPBF have very high computational costs. The developed PIDL method provides a unique avenue to overcome these challenges in these situations where one can do physics-based simulations for a very short period to get the model training data. Using that data one can find the model constants like *Re* and *Pe* in Eqs. (1–3). With the inferred model constants and training data of one process parameter (e.g., temperature), one may use the forward problem-solving method in Section 3 to efficiently predict the values of unknown parameters (e.g., velocity field) for future time steps. The fast convergence of the PIDL algorithm will significantly reduce the computation cost inherited with physics-based simulations.

In this study, the temperature and velocity data of 10,000 scattered points in the computation domain at 10 different time frames are input to the PIDL model for training. The expected output is the predicted value of the Reynolds number (*Re*), Peclet number (*Pe*), and their relative errors in Table 1. It shows a good agreement with the exact values. It should be mentioned that the PIDL model needs initial values for these model constants to learn.

Table 1
The inferred Reynolds number (*Re*) and Peclet number (*Pe*).

Inferred model constants	Training time: 5 hrs.		
	Reference	Inferred	Relative L_2 error
Re	100	105.38	5.38%
Pe	100	100.615	0.62%

5. Discussion and conclusions

This paper presents a data-efficient PIDL method to address the gas flow shear-driven multi-physical dynamics of the melt pool in LPBF. Several aspects of the PIDL method are worth discussing.

(1) The computation domain is a 2-dimensional (2D) rectangular zone for simplification. A more accurate domain geometry of the melt pool can be approximated based on a measurement of the cross-section of the solidified melt pool. The 2D domain can also be extended to a 3D domain.
(2) For the generalization purpose regardless of LPBF conditions, the temperature, velocity, and pressure parameters in the governing equations are normalized by their maximum values. This will simplify and streamline the PIDL model training. The predicted data in their normalized format can be conveniently converted to absolute values.
(3) Deep learning usually needs massive training data of high quality which are difficult to obtain in many processes. By embedding process physics, PIDL offers a unique advantage of strong generalization using small data.
(4) To infer an unknown model constant of the governing equations, the required initial value can be an averaged value or an estimated value from an experiment.

Unlike traditional data-driven ML models, the data-efficient PIDL model of the gas-melt pool dynamics generates the key results.

- The PIDL model may solve forward problems to predict the temperature and velocity fields under the impact of gas flow with only labeled discrete temperature data and boundary conditions.
- The PIDL model can solve inverse problems to learn the model constants of governing equations by using the scattered data of temperature and velocity in the melting domain.
- The PIDL algorithm is very robust as it functions very well on low spatial-temporal resolution.
- PIDL converges very fast by enforcing a soft penalty in the loss function for not satisfying the physical laws.

Supplemental dataset

The dataset and code used for the current study can be found at https://github.com/rhl8272/PIDL.git

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

Acknowledgment

The authors would like to thank the financial support of the National Science Foundation under the grant CMMI- 2152908.

References

[1] Hauser C, Childs THC, Dalgarno K, Eane R (1999) Atmospheric control during direct selective laser sintering of stainless steel 314S powder. In: *Proceedings of Solid Freeform Fabrication Symposium*, 265–272.
[2] Ferrar B, Mullen L, Jones E, Stamp R, Sutcliffe C (2012) Gas flow effects on selective laser melting (SLM) manufacturing performance. *Journal of Materials Processing Technology* 212:355–364.
[3] Caballero A, Suder W, Chen X, Pardal G, Williams S (2020) Effect of shielding conditions on bead profile and melting behaviour in laser powder bed fusion additive manufacturing. *Additive Manufacturing* 34:101342.
[4] Amano H, Ishimoto T, Suganuma R, Aiba K, Sun S, Ozasa R, Nakano T (2021) Effect of a helium gas atmosphere on the mechanical properties of Ti-6Al-4 V alloy built with laser powder bed fusion: a comparative study with argon gas. *Additive Manufacturing* 48:102444.
[5] Rabinovich E, Kalman H (2008) Generalized master curve for threshold superficial velocities in particle–fluid systems. *Powder Technology* 183(2):304–313.
[6] Ladewig A, Schlick G, Fisser M, Schulze V, Glatzel U (2016) Influence of the shielding gas flow on the removal of process by-products in the selective laser melting process. *Additive Manufacturing* 10:1–9.
[7] Wirth F, Frauchiger A, Gutknecht K, Cloots M (2021) Influence of the inert gas flow on the laser powder bed fusion (LPBF) process. in Meboldt M, Klahn C, (Eds.) *Industrializing Additive Manufacturing*, Springer, Cham, 192–204.
[8] Anwar A, Ibrahim I, Pham Q (2019) Spatter transport by inert gas flow in selective laser melting: a simulation study. *Powder Technology* 352:103–116.
[9] Guo S, Agarwal M, Cooper C, Tian Q, Gao R, Guo W, Guo YB (2021) Machine learning for metal additive manufacturing: towards a physics-informed data-driven paradigm. *Journal of Manufacturing Systems* 62:145–163.
[10] Zhu Q, Liu Z, Yan J (2021) Machine learning for metal additive manufacturing: predicting temperature and melt pool fluid dynamics using physics-informed neural networks. *Computational Mechanics* 67(2):619–635.
[11] Liao S., Xue T., Jeong J., Webster S., Ehmann K., Cao J. (2022) Hybrid full-field thermal characterization of additive manufacturing processes using physics-informed neural networks with data. arXiv:2206.07756.
[12] Tseng CC, Li CJ (2019) Numerical investigation of interfacial dynamics for the melt pool of Ti-6Al-4 V powders under a selective laser. *International Journal of Heat and Mass Transfer* 134:906–919.
[13] Raissi M, Yazdani A, Karniadakis GE (2020) Hidden fluid mechanics: learning velocity and pressure fields from flow visualizations. *Science* 367:1026–1030.