

A Model-Based Reinforcement Learning and Correction Framework for Process Control of Robotic Wire Arc Additive Manufacturing

Audelia G. Dharmawan, Yi Xiong, Shaohui Foong, and Gim Song Soh

Abstract—Robotic Wire Arc Additive Manufacturing (WAAM) utilizes a robot arm as a motion system to build 3D metallic objects by depositing weld beads one above the other in a layer by layer fashion. A key part of this approach is the process study and control of Multi-Layer Multi-Bead (MLMB) deposition, which is very sensitive to process parameters and prone to error stacking. Despite its importance, it has been receiving less attention than its single bead counterpart in literature, probably due to the higher experimental overhead and complexity of modeling. To address these challenges, this paper proposes an integrated learning-correction framework, adapted from Model-Based Reinforcement Learning, to iteratively learn the direct effect of process parameters on MLMB print while simultaneously correct for any inter-layer geometric digression such that the final output is still satisfactory. The advantage is that this learning architecture can be used in conjunction with actual parts printing (hence, in-situ study), thus minimizing the required training time and material wastage. The proposed learning framework is implemented on an actual robotic WAAM system and experimentally evaluated.

I. INTRODUCTION

Wire Arc Additive Manufacturing (WAAM) is a directed energy deposition manufacturing technique that builds metallic part layer by layer on a substrate using a motion system. Typically, it utilizes an electric arc as the energy source, wire as the feedstock, and an industrial robot arm as the motion system [1], [2]. Recently, this technique has gained increasing attention by both academia and industry for producing near net-shape large metallic parts, due to its high deposition rates and low buy-to-fly ratio [3]. The 3D parts are built by depositing overlapping welding beads in the horizontal (multi-bead) as well as vertical (multi-layer) directions. As such, each deposited layer acts as the substrate for the subsequent layer. Hence, it is important to ensure that the printed layer is of sufficient quality in order to provide a proper substrate for the subsequent layer's deposition [4], as irregular layer's surface commonly results in cumulative geometrical error leading to undesirable concave/convex surfaces as the print progresses vertically, as depicted in Fig. 1.

To optimize the flatness and uniformity of the deposited layer, various analytical formulations have been proposed to calculate the suitable stepover distance between the overlapping beads based on single bead process study [5], [6], [7], [8], [9]. However, they do not generalize well to Multi-Layer Multi-Bead (MLMB) scenario, as the process parameters

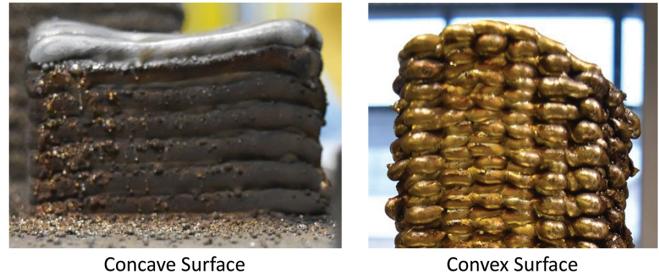


Fig. 1. Single bead study is typically not accurate enough to predict the output behaviour of a 3D print, commonly resulting in an accumulated error such as irregular or concave/convex surface finish as shown in the figure. On the other hand, multi-layer multi-bead study entails considerably higher overhead. This work proposes a cost-effective approach for multi-layer multi-bead process study through in-situ learning while printing actual parts, resulting in a better surface finish and more near-net-shape output.

designed based on the single bead models are not accurate enough for MLMB print due to the multitude of interacting process parameters [10]. To compensate for the inaccuracy of the single bead model for MLMB print, complex vision-based control has been employed to regulate the process parameters and the deposition in real-time to improve the quality of the print output [11], [12], [13]. However, implementing such a feedback control requires developing a complicated online measurement system, which is prone to noise and inaccuracies due to the intense welding arc. Alternatively, inter-layer milling has been explored to achieve the required surface flatness [14]. Such hybrid manufacturing method, despite being straightforward, still suffers from time and material wastage due to the added subtractive process, which diminishes the cost-effectiveness and merit of the WAAM manufacturing process.

It is to be noted that the MLMB aspect of WAAM has been rarely addressed and studied, probably due to i) the complexity of developing an accurate model that considers the variety of interacting process factors that affects the MLMB print behaviour [10]; and ii) the amount of experimental overhead, such as the time and material required, to study MLMB behaviour as compared to single bead, making the study more costly and less attractive. For instance, to study the influence of travel speed on MLMB print output, a number of sacrificial 3D hexahedrons needs to be printed and characterized [15]. This motivates us to explore an alternative approach of using machine learning towards the process study and control of MLMB print so as to improve the printed part consistency and accuracy, yet addressing the above two challengers.

The authors gratefully acknowledge the support of SUTD Digital Manufacturing and Design Centre (<https://dmnd.sutd.edu.sg>)

The authors are with the Singapore University of Technology and Design, Engineering Product Development Pillar, Singapore
 sohgimsong@sutd.edu.sg

In this paper, we propose an integrated learning-correction framework for MLMB print, adapted from model-based reinforcement learning. In the framework, the process model is iteratively learned and subsequently used to compensate for each layer's flatness errors “in situ”. The advantage is that this learning framework can be used in conjunction with the actual printing of parts (hence, in situ), minimizing the required upfront training time and material wastage. This work represents a preliminary study and is our first step towards an in-situ learning paradigm for robotic WAAM in order to boost MLMB process study to improve the resulting print quality, while at the same time still able to execute and deliver the manufacturing function to a large extent.

II. MODEL-BASED REINFORCEMENT LEARNING

In Reinforcement Learning (RL), an agent at each time step t , with state $s_t \in S$, executes some actions $a_t \in A$, receives a reward $r_t = r(s_t, a_t)$, and transits to the next state s_{t+1} according to some unknown dynamics function $f : S \times A \rightarrow S$. The goal is to learn a policy at each time step that takes actions which maximizes the sum of future rewards given by $\sum_{t'=t}^{\infty} \gamma^{t'-t} r_{t'}$, where $\gamma \in [0, 1]$ is a discount factor that prioritizes near-term rewards. This can be achieved with and without knowing the model of the environment dynamics, termed as model-based and model-free RL respectively, each with its own advantages and disadvantages.

Model-free RL has been shown to be capable of policy learning for a wide range of tasks. The downside is that it requires a very large number of samples for it to be effective. Model-based RL, on the other hand, is more sample efficient but requires some knowledge of the environment dynamics. As the aim of our learning framework for our in-situ WAAM process study requires the system to be able to learn with a few samples initially and subsequently learn an accurate process input-print output relationship, the model-based RL is adapted for our work.

In model-based RL [16], a model of the system dynamics is used to make predictions, which is subsequently used for action selection. Let $\hat{f}_{\theta}(s_t, a_t)$ denote a learned discrete-time dynamics function, parameterized by θ , that takes the current state s_t and action a_t and outputs an estimate of the next state at time $t + \Delta t$. The actions for H time steps ahead can be selected by solving the optimization problem:

$$(a_t, \dots, a_{t+H-1}) = \arg \max_{a_t, \dots, a_{t+H-1}} \sum_{t'=t}^{t+H-1} \gamma^{t'-t} r_{t'}. \quad (1)$$

The dynamics function \hat{f}_{θ} can be iteratively learned by alternating between gathering N new datapoints and retraining the model using the aggregated data in order to mitigate noisy observations and thus improve the performance of the model's prediction.

III. AN INTEGRATED LEARNING-CORRECTION FRAMEWORK

Figure 2 illustrates the proposed integrated learning-correction framework for WAAM process utilizing the Kriging

dynamics function. An agent in this framework denotes a discrete point on the print layer's path (waypoint). The state space encapsulates the observable print output behaviors (height, width, temperature, sound, etc.) and the action space comprises of the possible input process parameters (torch speed, wire feed rate, nozzle-to-substrate distance, torch angle, etc.). The common goal for all agents is to achieve a uniform surface height.

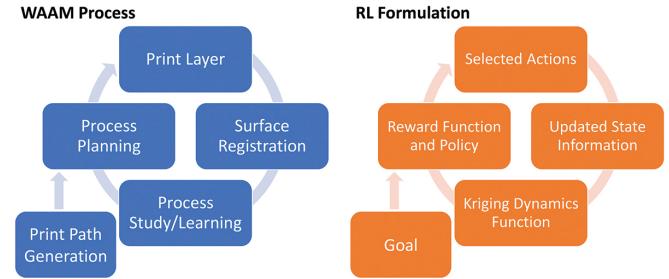


Fig. 2. Illustration of the proposed integrated learning-correction framework for WAAM process (left) and the corresponding RL expression (right).

In the following, we shall describe in more details how the agent's actions for the first iteration (i.e. first print layer) are initialized in section III-A, how the learned dynamics function is iteratively revised in section III-B, how the goal state for subsequent iterations is updated in section III-C, and how we extract a policy using the learned dynamics function and the reward function which directly incorporates the required correction in section III-D.

A. First Iteration Initialization

For the first iteration, random actions are usually taken and used to initialize the first training dataset. However, welding is a dangerous operation and it is unsafe to operate outside its acceptable process parameters. Thus, the action space is constrained to span within the weld process window, i.e. the lower and upper bounds of the weld process parameters, and they are distinct for different materials. Readers can refer to the single bead study [17] on how to establish such process window, which then can serve as the initial training dataset.

B. Learned Dynamics Function

1) *Training Dataset:* To learn the dynamics function, a set of training dataset needs to be established. As the print path is a continuous trajectory, discretizing it into waypoints provides us with multiple agents, each with its own local states and can be assigned with independent actions. Hence, we are also adopting a parallel RL framework [18], where the discrete points on the print path act as multiple agents learning the same task in parallel and pooling their experiences for the training update, which increases the learning rate. As such, the training dataset after each print layer is

$$D_t = \{(s_{1,t}, a_{1,t}, s_{1,t+1}), \dots, (s_{n_t,t}, a_{n_t,t}, s_{n_t,t+1})\}, \quad (2)$$

where n_t denotes the number of agents (discretized print points) at time step (layer) t . The agents can enter and leave (i.e. be updated) at every time step to accommodate for

changes in the print path between the layers for printing complex geometries. This method of generating the agents is then general enough for complicated print path such as in [19], [20].

2) *Kriging Dynamics Function:* In process modeling, neural network has been the commonly used approach for single bead process study [21], [22]. For our work, we parameterize the learned dynamics function \hat{f}_θ as Gaussian Process Regression (GPR) model [23], also known as the Kriging model. This model is known to provide better prediction in the presence of noisy observation and small dataset [13], [24]. We shall discuss briefly the Kriging model used here.

The GPR model is constructed based on the observed input-response pairs (\mathbf{X}, \mathbf{Y}) . The model predicts the response \mathbf{Y}^* of unevaluated input \mathbf{X}^* based on its locality with evaluated points in the input space. It assumes both observed and unobserved responses, \mathbf{Y} and \mathbf{Y}^* , have their finite-dimensional distributions as Gaussian, i.e. $\mathbf{Y} \sim GP(\mu_{\mathbf{Y}}, \Psi_{\mathbf{Y}})$ and $\mathbf{Y}^* \sim GP(\mu_{\mathbf{Y}^*}, \Psi_{\mathbf{Y}^*})$, where μ is the mean value and Ψ is the covariance function. Based on Bayes' Theorem, $P(\mathbf{Y}^* | \mathbf{Y}) = P(\mathbf{Y}, \mathbf{Y}^*) / P(\mathbf{Y})$, the Gaussian distributed P is expressed as

$$P(\mathbf{Y}^* | \mathbf{Y}) \sim GP(\mu_{\mathbf{Y}^*} + \Psi_{\mathbf{Y}, \mathbf{Y}^*}^T \Psi_{\mathbf{Y}}^{-1} (\mathbf{Y} - \mu_{\mathbf{Y}}), \Psi_{\mathbf{Y}^*} - \Psi_{\mathbf{Y}, \mathbf{Y}^*}^T \Psi_{\mathbf{Y}}^{-1} \Psi_{\mathbf{Y}, \mathbf{Y}^*}). \quad (3)$$

The set of mean values, μ , can be represented using a polynomial regression model $\beta \mathbf{H}$, where \mathbf{H} is a set of basis functions for the design parameters which can take any order and β is the corresponding coefficient vector and its prior is Gaussian $\beta \sim GP(\mathbf{b}, \mathbf{B})$. The best prediction $\hat{\mathbf{Y}}^*$ is

$$E(\mathbf{Y}^*) = \mathbf{H}_{\mathbf{Y}^*}^T \beta + \Psi_{\mathbf{Y}, \mathbf{Y}^*} \Psi_{\mathbf{Y}}^{-1} (\mathbf{Y} - \mathbf{H}_{\mathbf{Y}}^T \beta), \quad (4)$$

and the prediction variance is

$$\text{Var}(\mathbf{Y}^*) = \Psi_{\mathbf{Y}^*} - \Psi_{\mathbf{Y}, \mathbf{Y}^*} \Psi_{\mathbf{Y}}^{-1} \Psi_{\mathbf{Y}, \mathbf{Y}^*}^T + \mathbf{R}^T (\mathbf{B}^{-1} + \mathbf{H}_{\mathbf{Y}} \mathbf{K}_{\mathbf{Y}}^{-1} \mathbf{H}_{\mathbf{Y}}^T)^{-1} \mathbf{R}, \quad (5)$$

where $\beta = (\mathbf{B}^{-1} + \mathbf{H}_{\mathbf{Y}} \mathbf{K}_{\mathbf{Y}}^{-1} \mathbf{H}_{\mathbf{Y}}^T)^{-1} (\mathbf{H}_{\mathbf{Y}} \mathbf{K}_{\mathbf{Y}}^{-1} \mathbf{Y} + \mathbf{B}^{-1} \mathbf{b})$ and $\mathbf{R} = \mathbf{H}_{\mathbf{Y}}^* - \mathbf{H}_{\mathbf{Y}} \mathbf{K}_{\mathbf{Y}}^{-1} \mathbf{K}_{\mathbf{Y}, \mathbf{Y}^*}$.

Based on the Kriging model, we can learn a dynamics function that predicts the change in the agent's state \mathbf{s}_t under the action \mathbf{a}_t , i.e.

$$\hat{\mathbf{s}}_{t+1} = \mathbf{s}_t + \hat{f}_\theta(\mathbf{a}_t) \quad (6)$$

using the cumulative training dataset $D_T = \{D_1, \dots, D_t\}$, where t is the current time step.

C. Goal Specification

The goal defines the state that the agent needs to achieve. After printing layer t , the quality of the layer's surface and the necessary correction are quantified by scanning the top layer to obtain the surface point clouds $z_t(x, y)$. To update the goal specification for time step $t+1$, the next layer's print path is sliced and generated from the 3D CAD model based on the registered maximum height $z_{t,max}$ of the scanned

layer [25]. A simple alternating direction strategy is applied to the print path to mitigate the effect of arc striking and extinguishing [26]. The common goal of all the agents is then to achieve a uniform surface height,

$$s_{t+1,d}^z = z_{t,max} + l, \quad (7)$$

where l is the expected print height increment after printing the next layer.

D. Reward Function and Policy

The reward function is formulated such that the agent is penalized for selecting the actions that are predicted to result in a deviation from the desired goal state. The agent is rewarded with a weighted κ amount of the estimated standard deviation of the prediction σ_θ from the learned dynamics function to encourage slight exploration especially at the initial learning iterations. The reward function for each agent i is formulated as

$$r(\mathbf{s}_{i,t}, \mathbf{a}_{i,t}) = -|s_{i+1,d}^z - \hat{f}_\theta(\mathbf{a}_{i,t}) - s_{i,t}^z| + \kappa \sigma_\theta(\mathbf{a}_{i,t}). \quad (8)$$

By also incorporating the agent's current height state $s_{i,t}^z$ in the reward function, each agent is thus encouraged to select the actions that achieve the next goal state and simultaneously correct its own current deviation from the previous goal state. With the formulated reward function, each agent i then selects the actions that maximizes the reward based on a greedy policy, i.e.

$$\mathbf{a}_{i,t} = \arg \max_{\mathbf{a}_{i,t}} r(\mathbf{s}_{i,t}, \mathbf{a}_{i,t}). \quad (9)$$

Solving Eqn. (9) analytically is difficult due to the learned dynamics function being nonlinear. To obtain approximate solutions, we employ non-probability systematic sampling where K candidate action sets are generated starting from the lower limit of the action window with a fixed sampling interval up to the upper limit. The corresponding states are then predicted using the learned dynamics function, the rewards are calculated, and the candidate action set with the highest expected reward is selected.

Algorithm 1 summarizes the model-based parallel reinforcement learning approach for the proposed in-situ process study and control of WAAM. In the event of printing a brand new component but continuing the learning, line 1 and line 2 can be omitted.

IV. EXPERIMENTAL SETUP

To demonstrate and evaluate the potential and feasibility of the proposed integrated learning-correction framework for process study and control, we implemented the algorithm on our robotic WAAM system developed at Singapore University of Technology and Design (SUTD) as shown in Fig. 3. The system consists of a robot manipulator (ABB IRB 1660ID), a welding power source (Fronius TPS 400i) equipped with a welding torch (Fronius WF 25i Robacta Drive), a cartesian coordinate robot made up of three linear rails (PMI KM4510) powered by three servos (SmartMotor SM34165DT), and a 2D laser scanner (Micro-Epsilon scanCONTROL 2910-100). The gantry system is controlled to

Algorithm 1 In-Situ Process Reinforcement Learning

- 1: Gather initial dataset D_0 from the single bead process window
- 2: Initialize empty dataset D_T
- 3: **for** $t = 1$ to max_layer **do**
- 4: Train \hat{f}_θ using D_0 and D_T
- 5: Update goal $s_{t+1,d}^z$, and generate n_t agents
- 6: **for** $i = 1$ to n_t **do**
- 7: Get agent's current state $s_{i,t}$
- 8: Use \hat{f}_θ to estimate the optimal action $a_{i,t}$
- 9: Execute action $a_{i,t}$
- 10: Get agent's next state $s_{i,t+1}$
- 11: Add $(s_{i,t}, a_{i,t}, s_{i,t+1})$ to D_T
- 12: **end for**
- 13: **end for**

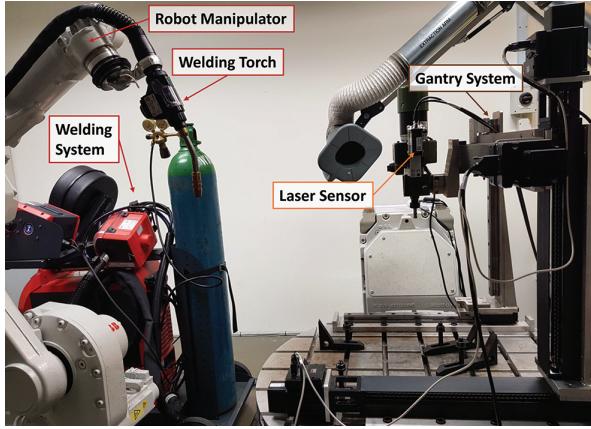


Fig. 3. The Robotic Wire Arc Additive Manufacturing (WAAM) System developed at Singapore University of Technology and Design (SUTD).

move the line laser scanner in 3D space in order to obtain 3D point clouds of the printed layer's surface.

For the preliminary evaluation of the proposed learning framework, we formulated torch speed and wire feed rate as the agent's actions as well as print height as the observed agent's state, as they are the key variables and parameters that are known to influence the print behaviour and are crucial to regulate [27]. The agent's local state is obtained from the laser scan output of the print surface by taking the mean value of the print height within a radius δ mm from the agent, as depicted in Fig. 4.

To demonstrate the robustness and adaptability of the approach, we carried out the experiments with two different metals, bronze (ERCuNiAl) and stainless steel (ER316LSi). For the bronze material, two hexahedrons of size $50 \times 50 \times 50$ mm were printed, one using the proposed learning-correction framework and one using the process parameters recommended by the single bead study, in order to directly compare the effect of the proposed learning framework. For the stainless steel material, one hexahedron was printed using the process parameters recommended by the single bead study for result comparison, while a more complex actual part representing the shape of a twistlock pin, having

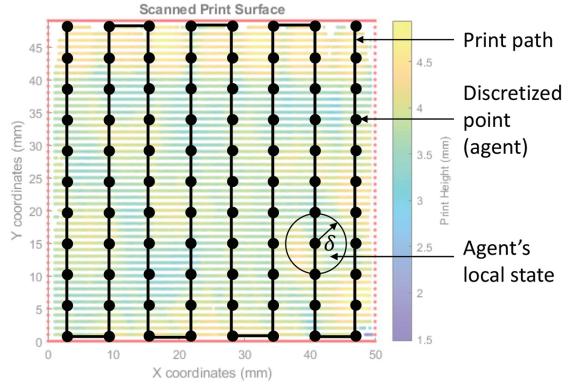


Fig. 4. In the proposed framework, each discretized point on the layer's print path acts as an agent with local states and independent actions, performing parallel model-based reinforcement learning with correction. The agent's local state is obtained by taking the mean of the observation within δ unit from the agent.

a total height of 460 mm and several different deposition paths throughout the print, was printed using the proposed learning-correction framework up to a height of 360 mm to demonstrate the possibility of using the proposed learning framework for printing actual components with varying print paths and attaining a more neat-net-shape output. The remaining 100 mm of the twistlock pin was printed without using the framework in order to directly compare the output without wasting materials.

V. EXPERIMENTAL RESULTS

Prior to printing the parts, single bead study was performed to establish the process window, determine the commonly recommended process parameters based on the single bead study using the method in [6], as well as collect a few data to initialize the learned dynamics function. Figure 5 shows samples of the output of the performed single bead study. For the single bead study, several welding beads were printed using different process parameters. The beads were then scanned using the moving 2D laser scanner. The point cloud data was first filtered using a moving average filter and the weld bead's toe points were extracted from the second derivative of the filtered data. Based on the single bead study, the process window that was selected for the experiments was torch speed of [6, 10] mm/s and wire feed rate of [6, 7] m/min for bronze, as well as torch speed of [7, 13] mm/s and wire feed rate of [3, 5] m/min for stainless steel.

A. Bronze Material

For the bronze experiments, one hexahedron was printed using the parameters recommended by the single bead study outcome, while another hexahedron was printed by implementing the proposed learning framework as in Alg. 1. Serving as an illustration, Fig. 6 shows sample of the selected actions using the algorithm based on the agent's local state. The individual agent selects the optimal actions to achieve the goal state given its current state through Eqn. (9).

Figure 7 then shows the final output of the printed parts. As seen from the photograph, the print implementing

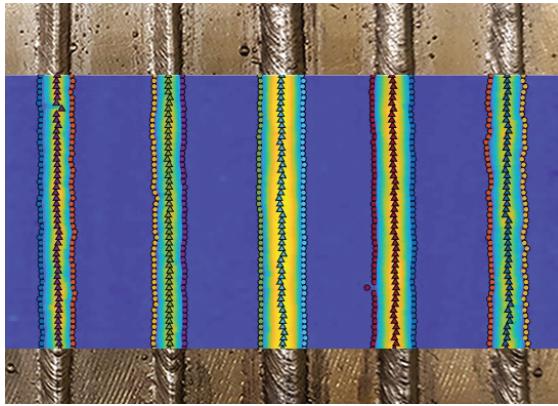


Fig. 5. Photograph of the single bead study, overlaid with the analyzed point clouds to extract the data.

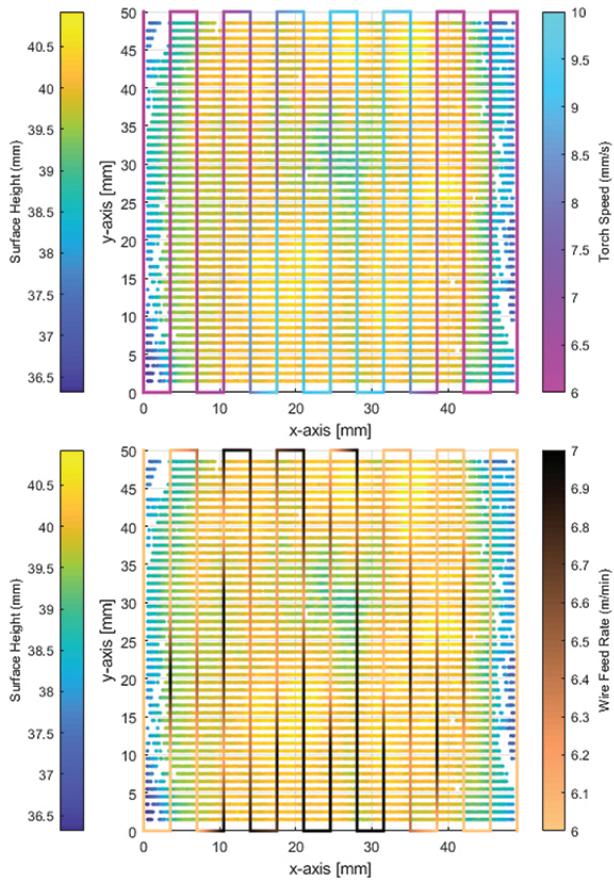


Fig. 6. Example of the selected independent actions based on the agent's local state in order to achieve the goal of uniform surface height using the proposed learning framework.

the proposed integrated learning-correction framework (left hexahedron) results in a more uniform surface height and thus a more near-net-shape output. The favourable output of the prints signifies the potential of the proposed learning framework for in-situ process learning while simultaneously printing actual parts, which will be further demonstrated by the output of the stainless steel actual part's print.

Figure 8 further illustrates and details the layer evolution



Fig. 7. Print outputs of bronze: using the proposed learning framework (left), and using the single-bead recommended parameters (right).

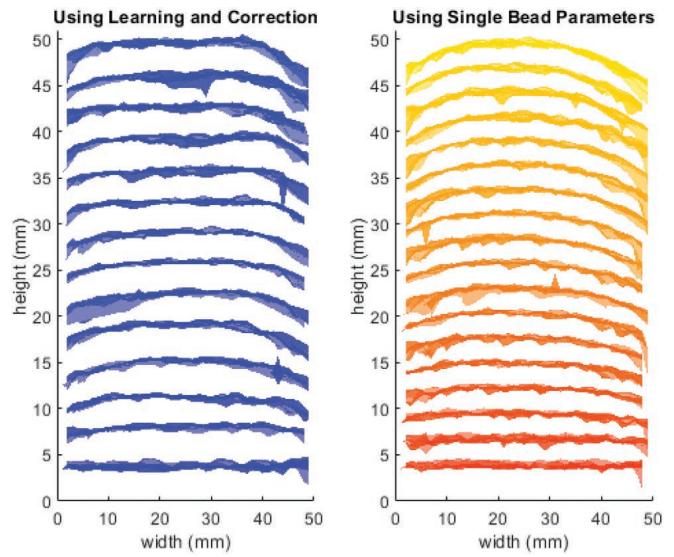


Fig. 8. Illustration of the layer evolution of the prints reconstructed from the laser scan output. Refer to Fig. 10 for the standard deviation.

of both prints constructed from the laser scan output of each layer. As seen from the reconstructed surface from the point clouds, the print using the proposed learning framework is able to produce a more uniform layer's surface height throughout the print layers, while the print using the single-bead recommended parameters accumulates error as the print height progresses resulting in a more convex surface. At the worst case, the accumulated error will lead to print failure.

B. Stainless Steel Material

For stainless steel material, one hexahedron was printed using the parameters recommended by the single bead study for result comparison, while a more complex actual part representing the shape of a twistlock pin with a height of 460 mm was printed using the proposed learning framework up to a height of 360 mm and the remaining 100 mm without the framework to directly compare the print output as shown in Fig. 9. As seen from the photograph, the print implementing the proposed integrated learning-correction framework (left) results in a flat surface, while the print without using the framework (right) exhibits a deep valley which keeps accumulating as the print height increases.



Fig. 9. Print output of stainless steel twistlock pin actual component: using the proposed learning framework up to 360 mm height (left), and using the single-bead recommended parameters for the remaining 100 mm (right).

C. Discussion

To further quantitatively compare the layer's surface uniformity of the prints, the standard deviation (STD) of the surface height of each printed layer is calculated from the layer's surface scan output and the values are plotted in Fig. 10 for the bronze material and Fig. 11 for the stainless steel prints. As seen from the plot, the standard deviation of the layer's surface height printed using the recommended single-bead parameters has an increasing trend as the print height progresses vertically for both materials, depicting an accumulation of error. The prints implementing the proposed learning framework initially produces larger surface standard deviation, most likely due to the initial learning process where the system is just starting to explore and evaluate the influence of the different manufacturing parameters on the MLMB print behaviour. As the print layer and the learning progresses, the system is then able to have a better understanding of the manufacturing process and select the optimal parameters to correct its previous error as well as achieving the goal, resulting in a more bounded and lower standard deviation and thus a more near-net-shape output. For the more complex twistlock pin print using the learning framework, the surface standard deviation is still much lower and bounded even at a height of seven times the simple hexahedron print without using the framework, as shown in Fig. 11. The results of the preliminary experiments on the two different materials provide a positive feedback on the feasibility of using the proposed learning framework to perform in-situ process study and correction.

VI. CONCLUSIONS

This paper presents an integrated model-based reinforcement learning-correction framework for in-situ MLMB process learning of robotic WAAM, as well as the preliminary experimental study of the learning framework on a physical robotic WAAM system performing printing tasks for two different materials. From the experimental results, the print outputs obtained using the proposed learning framework exhibit better surface finish and more near-net shape. This demonstrates the feasibility of the formulated learning architecture for in-situ process learning and control. The

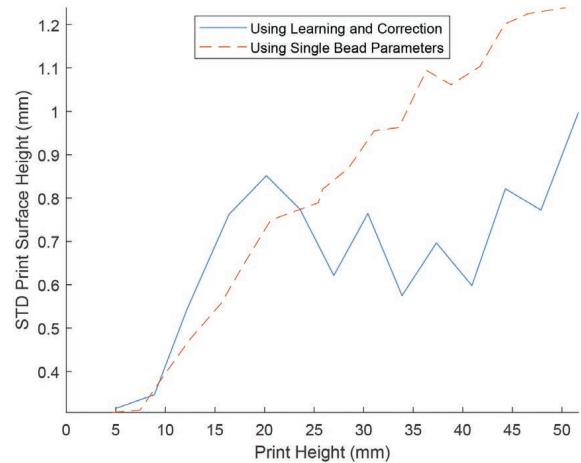


Fig. 10. Comparison of the standard deviation (STD) of the layers' surface finish of bronze material between the print using the learning-correction framework and the recommended parameters from single-bead study.

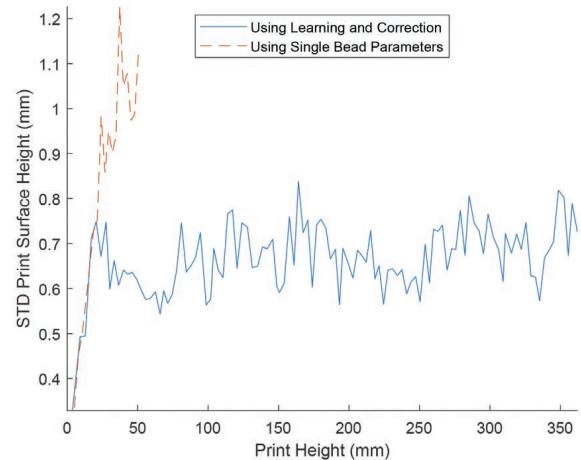


Fig. 11. Comparison of the standard deviation (STD) of the layers' surface finish of stainless steel between the print using the learning-correction framework and the recommended parameters from single-bead study.

encouraging result of this study opens up the possibility to perform cost-effective MLMB process learning, an aspect of WAAM that has been rarely studied despite being important due to its high experimental overhead cost and complexity of modeling.

Based on this positive preliminary outcome, our future work seeks to extend the proposed learning framework to a comprehensive in-situ process learning. Additional process parameters, observed behaviours with multiple sensors, as well as goal specifications will be added into the learning architecture and evaluated to achieve a more in-depth process learning for an even better print. Indeed, different formulation can also be constructed into the learning framework, such as treating a layer's print path as a trajectory of an agent instead of discretizing it into multiple agents, and it will be interesting to analyze the performance of the different formulations on the learned prediction and output. It is of interest also in the future to study transfer learning between the different materials to further speed up the learning.

REFERENCES

- [1] S. W. Williams, F. Martina, A. C. Addison, J. Ding, G. Pardal, and P. Colegrove, "Wire+ arc additive manufacturing," *Materials Science and Technology*, vol. 32, no. 7, pp. 641–647, 2016.
- [2] A. G. Dharmawan, S. Padmanathan, Y. Xiong, I. F. Ituarte, S. Foong, and G. S. Soh, "Maximizing robot manipulators functional redundancy via sequential informed optimization," in *2018 3rd International Conference on Advanced Robotics and Mechatronics (ICARM)*. IEEE, 2018, pp. 334–339.
- [3] A. Busachi, J. Erkoyuncu, P. Colegrove, F. Martina, and J. Ding, "Designing a waam based manufacturing system for defence applications," *Procedia Cirp*, vol. 37, pp. 48–53, 2015.
- [4] S. Tang, G. Wang, and H. Zhang, "In situ 3d monitoring and control of geometric signatures in wire and arc additive manufacturing," *Surface Topography: Metrology and Properties*, vol. 7, no. 2, p. 025013, 2019.
- [5] W. Aiyiti, W. Zhao, B. Lu, and Y. Tang, "Investigation of the overlapping parameters of mpaw-based rapid prototyping," *Rapid Prototyping Journal*, 2006.
- [6] S. Suryakumar, K. Karunakaran, A. Bernard, U. Chandrasekhar, N. Raghavender, and D. Sharma, "Weld bead modeling and process optimization in hybrid layered manufacturing," *Computer-Aided Design*, vol. 43, no. 4, pp. 331–344, 2011.
- [7] Y. Cao, S. Zhu, X. Liang, and W. Wang, "Overlapping model of beads and curve fitting of bead section for rapid manufacturing by robotic mag welding process," *Robotics and Computer-Integrated Manufacturing*, vol. 27, no. 3, pp. 641–645, 2011.
- [8] D. Ding, Z. Pan, D. Cuiuri, and H. Li, "A multi-bead overlapping model for robotic wire and arc additive manufacturing (waam)," *Robotics and Computer-Integrated Manufacturing*, vol. 31, pp. 101–110, 2015.
- [9] Y. Li, Y. Sun, Q. Han, G. Zhang, and I. Horváth, "Enhanced beads overlapping model for wire and arc additive manufacturing of multi-layer multi-bead metallic parts," *Journal of Materials Processing Technology*, vol. 252, pp. 838–848, 2018.
- [10] Y. Li, Q. Han, G. Zhang, and I. Horváth, "A layers-overlapping strategy for robotic wire and arc additive manufacturing of multi-layer multi-bead components with homogeneous layers," *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 9-12, pp. 3331–3344, 2018.
- [11] J. Xiong, G. Zhang, Z. Qiu, and Y. Li, "Vision-sensing and bead width control of a single-bead multi-layer part: material and energy savings in gmaw-based rapid manufacturing," *Journal of Cleaner Production*, vol. 41, pp. 82–88, 2013.
- [12] J. Xiong and G. Zhang, "Adaptive control of deposited height in gmaw-based layer additive manufacturing," *Journal of Materials Processing Technology*, vol. 214, no. 4, pp. 962–968, 2014.
- [13] H. Dong, M. Cong, Y. Zhang, Y. Liu, and H. Chen, "Real time welding parameter prediction for desired character performance," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1794–1799.
- [14] Y.-A. Song, S. Park, D. Choi, and H. Jee, "3d welding and milling: Part i—a direct approach for freeform fabrication of metallic prototypes," *International Journal of Machine Tools and Manufacture*, vol. 45, no. 9, pp. 1057–1062, 2005.
- [15] I. Garmendia, J. Pujana, A. Lamikiz, J. Flores, and M. Madarieta, "Development of an intra-layer adaptive toolpath generation control procedure in the laser metal wire deposition process," *Materials*, vol. 12, no. 3, p. 352, 2019.
- [16] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7559–7566.
- [17] M. Somashekara and S. Suryakumar, "Studies on dissimilar twin-wire weld-deposition for additive manufacturing applications," *Transactions of the Indian Institute of Metals*, vol. 70, no. 8, pp. 2123–2135, 2017.
- [18] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 3389–3396.
- [19] H. Zhao, F. Gu, Q.-X. Huang, J. Garcia, Y. Chen, C. Tu, B. Benes, H. Zhang, D. Cohen-Or, and B. Chen, "Connected fermat spirals for layered fabrication," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–10, 2016.
- [20] C. Dai, C. C. Wang, C. Wu, S. Lefebvre, G. Fang, and Y.-J. Liu, "Support-free volume printing by multi-axis motion," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.
- [21] J. Xiong, G. Zhang, J. Hu, and Y. Li, "Forecasting process parameters for gmaw-based rapid manufacturing using closed-loop iteration based on neural network," *The International Journal of Advanced Manufacturing Technology*, vol. 69, no. 1-4, pp. 743–751, 2013.
- [22] D. Ding, Z. Pan, D. Cuiuri, H. Li, S. van Duin, and N. Larkin, "Bead modelling and implementation of adaptive mat path in wire and arc additive manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 39, pp. 32–42, 2016.
- [23] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.
- [24] Y. Xiong, S.-I. Park, S. Padmanathan, A. G. Dharmawan, S. Foong, D. W. Rosen, and G. S. Soh, "Process planning for adaptive contour parallel toolpath in additive manufacturing with variable bead width," *The International Journal of Advanced Manufacturing Technology*, pp. 1–12, 2019.
- [25] A. G. Dharmawan, Y. Xiong, S. Foong, and G. S. Soh, "Development of an automated and adaptive system for robotic hybrid-wire arc additive manufacturing (h-waam)," in *IFToMM International Symposium on Robotics and Mechatronics*. Springer, 2019, pp. 323–333.
- [26] J. Xiong, Z. Yin, and W. Zhang, "Forming appearance control of arc striking and extinguishing area in multi-layer single-pass gmaw-based additive manufacturing," *The International Journal of Advanced Manufacturing Technology*, vol. 87, no. 1-4, pp. 579–586, 2016.
- [27] Z. Hu, X. Qin, T. Shao, and H. Liu, "Understanding and overcoming of abnormality at start and end of the weld bead in additive manufacturing with gmaw," *The International Journal of Advanced Manufacturing Technology*, vol. 95, no. 5-8, pp. 2357–2368, 2018.