

000 PHYSICS-INFORMED NEURAL NETWORKS WITH 001 002 MESSAGE-PASSING WEIGHTS 003 004

005 **Anonymous authors**

006 Paper under double-blind review

007 008 ABSTRACT 009

011 Adaptive loss balancing algorithms play a crucial role in improving the performance of Physics-Informed Neural Networks (PINNs) by effectively managing
012 the weights assigned to different loss components. Most notably, Wang et al.
013 (2022a) introduced Causal Physics-Informed Neural Networks (Causal PINNs),
014 which achieve superior performance by simply reformulating the loss function
015 based on the causal structure that emerges from time dependency. However, de-
016 spite their empirical success, a solid theoretical analysis for the effectiveness of
017 Causal PINNs has not received adequate attention. This paper addresses this gap
018 by providing a theoretical rationale for Causal PINNs through the Belief Propa-
019 gation (BP) algorithm, which is commonly used for causal inference. In addition,
020 motivated by this analysis, we propose a Message Passing PINNs (MP-PINNs),
021 a novel adaptive weighting algorithm. Through extensive numerical experiments,
022 we demonstrate that the proposed MP-PINNs significantly outperform existing
023 adaptive weighting methods, exhibiting superior performance in solving complex
024 PDEs. Our findings highlight the potential of MP-PINNs as a powerful tool to
025 enhance both the accuracy and efficiency of PINNs.

026 027 1 INTRODUCTION

030 Physics-Informed Neural Networks (PINNs) have emerged as a powerful framework for solving
031 scientific and engineering problems by integrating physical laws directly into the learning process.
032 These networks leverage the principles of physics to guide the training of neural networks, allowing
033 them to learn not only from data but also from the underlying governing equations of the phenomena
034 being modeled. This innovative approach has been successfully applied to various domains, includ-
035 ing fluid dynamics, structural mechanics, and heat transfer, demonstrating significant improvements
036 in accuracy and generalization capabilities compared to traditional data-driven methods Raissi et al.
037 (2019); Karniadakis et al. (2021).

038 Adaptive loss balancing algorithms are essential for enhancing the performance of PINNs by man-
039 aging the weights assigned to different loss components . These algorithms help mitigate potential
040 imbalances that may arise during the training process, thereby improving the overall efficiency of
041 data-driven learning that incorporates physical principles. PINNs have demonstrated considerable
042 promise in addressing a wide range of scientific and engineering challenges, making them valuable
043 tools in these fields.

044 Causal Physics-Informed Neural Networks (Causal PINNs) represent a significant advancement in
045 this area, achieving superior performance by explicitly reformulating the loss function to respect
046 causality Wang et al. (2022a). However, despite their empirical success, a solid theoretical justifi-
047 cation for the effectiveness of Causal PINNs has been lacking. The absence of a robust theoretical
048 foundation may limit the applicability and extension of these approaches, posing a critical barrier to
049 the advancement of PINNs.

050 The Belief Propagation (BP) algorithm is a message-passing (MP) algorithm used for causal infer-
051 ence. For instance, in certain scientific fields, if causality is established between instances, BP can
052 be utilized to infer their exact causal relationships Chang et al. (2014). Also the mechanism of the
053 MP algorithm is already widely applied in machine learning and deep learning areas (Scarselli et al.,
2008; Gilmer et al., 2017; 2020). However, there have been no attempts to connect this to adaptive

054 weighting algorithms for PINNs. We note that Causal PINNs only consider time-dependency, al-
 055 though spatial dependency may also influence the calculate of the residual loss. Furthermore, Daw
 056 et al. (2022) discussed how information from the initial and boundary conditions can propagate to
 057 interior collocation points. In this context, the BP algorithm may suggests a method for analyzing
 058 the causal relationships between each collocation point.

059 This paper addresses the gap between the performance of Causal PINNs and the lack of analyt-
 060 ical understanding by providing a theoretical reformulation of their algorithms based on the BP
 061 algorithm. Furthermore, building on this analysis, we propose a novel adaptive weighting algo-
 062 rithm, termed Message Passing PINNs (MP-PINNs). Through extensive numerical experiments,
 063 we demonstrate that the proposed MP-PINNs significantly outperform existing adaptive weighting
 064 methods, exhibiting superior performance in solving complex partial differential equations (PDEs).
 065 Our findings highlight the potential of MP-PINNs as a powerful tool for enhancing both the accuracy
 066 and efficiency of PINNs.

067 The main contributions of this work can be summarized as follows:
 068

- We reframe Causal PINNs from the perspective of the BP algorithm
- Inspired by this analysis, we propose a novel adaptive weighting algorithm termed Message-Passing PINNs (MP-PINNs)
- We demonstrate, through numerical experiments, that the proposed MP-PINNs signifi-
 074 cantly outperform existing adaptive weighting algorithms.

076 1.1 RELATED WORKS

078 **Failure modes of PINNs** Despite the applicability of PINNs for solving various PDEs, there are
 079 still various problems to solve. Escepically, (Krishnapriyan et al., 2021) presented a possible failure
 080 mode of PINN, describing the challenging loss landscape that can appear in time-dependent PDEs.
 081

082 **Adaptive weights of PINNs** One strategy to improve PINN training is the use of adaptive weighting
 083 methods. For example, McClenney & Braga-Neto (2020) introduced a self-adaptive approach based
 084 on a soft attention mechanism, where weights are trained adversarially. Son et al. (2023) framed
 085 PINNs as a constrained optimization problem and applied the augmented Lagrangian method to de-
 086 fine an unconstrained minimax problem. By solving this using gradient descent and ascent, they
 087 demonstrated the effectiveness of a loss-balancing algorithm. Additionally, Wang et al. (2020) pro-
 088 posed a balancing method that directly influences gradient statistics, while Wang et al. (2022b)
 089 explored the Neural Tangent Kernel of PINNs to further enhance training.

090 **Causal PINNs** To address the failure modes of PINNs, (Wang et al., 2022a; Penwarden et al., 2023)
 091 defined this issue as the failure to deliver accurate information to collocation points that are distant
 092 from the initial points. Furthermore, they employed weighted loss to train PINNs, using namely
 093 causal weights. However this approach only considers the causality of time dependency rather than
 094 spatial dependency.

095 2 BACKGROUND

098 2.1 CAUSAL WEIGHTS FOR TRAINING PINNs

100 The causal PINN is based on the *causal weights* which is defined on the time-dependent PDEs
 101 (Wang et al., 2022a). For details, they consider the specific form of PDE given as

$$102 \quad \mathcal{R}[u](t, x) := \partial_t u(t, x) + \mathcal{N}[u](t, x) = 0, \quad t \in [0, T], x \in \Omega$$

104 subject to the initial and boundary conditions
 105

$$106 \quad \begin{aligned} \mathcal{I}[u](0, x) &= 0, & x \in \Omega, \\ 107 \quad \mathcal{B}[u](t, x) &= 0, & t \in [0, T], x \in \partial\Omega \end{aligned}$$

108 where $N[\cdot]$ is a linear or nonlinear differential operator, $B[\cdot]$ is a boundary operator and u is a
 109 unknown solution. To solve this, PINN loss generally defined as follows:
 110

$$111 \quad \mathcal{L}(\theta) = \lambda_{ic}\mathcal{L}_{ic}(\theta) + \lambda_{bc}\mathcal{L}_{bc}(\theta) + \mathcal{L}_r(\theta), \quad (1)$$

$$112 \quad \mathcal{L}_{ic}(\theta) = \frac{1}{N_{ic}} \sum_{i=1}^{N_{ic}} |\mathcal{I}[u_\theta](0, x_{ic}^i)|^2, \quad (2)$$

$$115 \quad \mathcal{L}_{bc}(\theta) = \frac{1}{N_{bc}} \sum_{i=1}^{N_{bc}} |\mathcal{B}[u_\theta](t_{bc}^i, x_{bc}^i)|^2 \quad (3)$$

118 and the residual loss \mathcal{L}_r of general equation is defined as the weighted sum over fixed time grids
 119 t_1, \dots, t_{N_t} dividing $[0, T]$, using causal weights given as follows:
 120

$$121 \quad \mathcal{L}_r(\theta) = \sum_{i=1}^{N_t} w_i(\theta) \mathcal{L}_r(t_i; \theta),$$

$$124 \quad \mathcal{L}_r(t; \theta) = \frac{1}{N_x} \sum_{j=1}^{N_x} |\mathcal{R}[u_\theta](t, x_j)|^2,$$

$$127 \quad w_i(\theta) = \exp(-\epsilon \sum_{k=1}^{i-1} \mathcal{L}_r(t_k; \theta))$$

130 where *causality parameter* ϵ determines the steepness of the weight w_i , which is controlled through
 131 an annealing strategy involving an increasing sequence of ϵ values.
 132

133 2.2 BELIEF-PROPAGATION ALGORITHM

135 In this section, we introduce the Belief Propagation (BP) with sum–product message passing which
 136 is used to infer a marginal probability for each single random variables over a given Markov random
 137 field. More specifically, we only consider the case that every potential function ψ related to each
 138 factor of the random field has two input variables, i.e. the joint probability has the form

$$139 \quad \mathbb{P}(X_i = x_i, \forall i \in V) = \prod_{\{i,j\} \in E} \psi_{i,j}(x_i, x_j).$$

142 with random variables $X_i : S_i \rightarrow \mathbb{R}$ corresponds to each nodes i of given factor graph $G =$
 143 (V, E, ψ) . The BP algorithm for marginal inference is consisting of two stages: (1) *message-passing*
 144 process MP and (2) *normalization* process NM . Through these stages, the *message* $\mu_{i \rightarrow j} :=$
 145 $\mu_{\{i,j\} \rightarrow j} : S_j \rightarrow [0, 1]$ from factor $\{i, j\} \in E$ to node $j \in V$ is calculated iteratively, i.e. $\mu^{(\tau+1)} :=$
 146 $\{\mu_{i \rightarrow j}^{(\tau+1)}\}_{\{i,j\} \in E} = (NM \circ MP)(\mu^{(\tau)})$ for each iteration step τ . First, the message-passing process
 147 is as follows:

$$148 \quad MP(\mu^{(\tau)})_{i \rightarrow j}(x_j) = \sum_{x_i \in S_i} \psi_{i,j}(x_i, x_j) \prod_{k \in N_i \setminus \{j\}} \mu_{k \rightarrow i}^{(\tau)}(x_i), \quad (4)$$

150 for each $\{i, j\} \in E$, $x_j \in S_j$ and iteration τ where N_j denotes the neighborhood of j , i.e. $N_j :=$
 151 $\{v \in V : \{j, v\} \in E\}$. Next, for the normalization process, All messages are normalized in L1
 152 sense, i.e.

$$154 \quad \mu_{i \rightarrow j}^{(\tau+1)}(x_j) := NM(MP(\mu^{(\tau)}))_{i \rightarrow j} = \frac{MP(\mu^{(\tau)})_{i \rightarrow j}(x_j)}{\sum_{x \in S_j} MP(\mu^{(\tau)})_{i \rightarrow j}(x)} \text{ for each } x_j \in S_j.$$

156 Note that when G has no loops, then the BP algorithm naturally requires only a single iteration.
 157 However, if there's a loop, then it conditionally converges. After τ iterations, the marginal inference
 158 for $\mathbb{P}(X_i = x_i)$ for each X_i is conducted by calculating *belief* $b_i^{(\tau)} : S_i \rightarrow [0, 1]$ for each node i as
 159

$$160 \quad \mathbb{P}(X_i = x_i) \approx b_i^{(\tau)}(x_i) = \frac{\prod_{j \in N(i)} \mu_{j \rightarrow i}^{(\tau)}(x_i)}{\sum_{x \in S_i} \prod_{j \in N(i)} \mu_{j \rightarrow i}^{(\tau)}(x)} \text{ for each } x_i \in S_j.$$

162 3 THE CONNECTION BETWEEN THE BP ALGORITHM AND CAUSAL
 163 WEIGHTS
 164

165 In this section, we will demonstrate how the causal weights can be connected to the causal inference
 166 over a Markov random field and derived using the BP algorithm. In the formulation of 2.1, the
 167 factor graph $G = (V, E, \psi)$ is defined as:

$$168 \quad V = \{v_i : i = 1, \dots, N_t\}$$

$$169 \quad E = \{e_i := (v_i, v_{i+1}) : i = 1, \dots, N_t - 1\}$$

170 based on the collocation points (t_i, x_j) for training PINN $u_\theta(t, x)$ with trainable parameter θ . Furthermore, for each random variable $X_i := X_{v_i} : S_i \rightarrow \mathbb{R}$ related to the node $v \in V$, the sample space S_i is defined as $\{-1, 1\}$, and the joint probability of $\{X_i\}_{i=1}^{N_t}$ is defined as

$$174 \quad \mathbb{P}(X_i = x_i, i = 1, \dots, N_t; \theta) = \prod_{i=1}^{N_t-1} \psi_{e_i}(x_i, x_{i+1}; \theta)$$

177 where the potential function $\psi_{e_i} : S_i \times S_{i+1} \rightarrow [0, \infty)$ is defined as

$$178 \quad \psi_{e_i}(x_i, x_{i+1}; \theta) = \begin{cases} \exp(-\epsilon \mathcal{L}_r(t_i; \theta)), & \text{for } x_i = 1, x_{i+1} = 1, \\ 1 - \exp(-\epsilon \mathcal{L}_r(t_i; \theta)), & \text{for } x_i = 1, x_{i+1} = -1, \\ 0, & \text{for } x_i = -1, x_{i+1} = 1, \\ 1, & \text{for } x_i = -1, x_{i+1} = -1 \end{cases}$$

183 for each $i = 1, \dots, N_t - 1$.

184 To find the marginal $\mathbb{P}(X_i)$ for each X_i , the message-passing process of BP algorithm runs as
 185 follows:

$$186 \quad \mu_{i \rightarrow i+1}(x_{i+1}) = \begin{cases} 1 \cdot \mu_{i-1 \rightarrow i}(-1) + (1 - \exp(-\epsilon \mathcal{L}_r(t_i; \theta))) \cdot \mu_{i-1 \rightarrow i}(1), & (x_{i+1} = -1) \\ 0 \cdot \mu_{i-1 \rightarrow i}(-1) + \exp(-\epsilon \mathcal{L}_r(t_i; \theta)) \cdot \mu_{i-1 \rightarrow i}(1) & (x_{i+1} = 1) \end{cases}$$

189 for $i = 2, \dots, N_t - 1$ and

$$190 \quad \mu_{i+1 \rightarrow i}(x_i) = \begin{cases} 1 \cdot \mu_{i+2 \rightarrow i+1}(-1) + 0 \cdot \mu_{i+2 \rightarrow i+1}(1), & (x_i = -1) \\ (1 - \exp(-\epsilon \mathcal{L}_r(t_i; \theta))) \cdot \mu_{i+2 \rightarrow i+1}(-1) + \exp(-\epsilon \mathcal{L}_r(t_i; \theta)) \cdot \mu_{i+2 \rightarrow i+1}(1) & (x_i = 1) \end{cases}$$

192 for $i = 1, \dots, N_t - 2$ where the boundary conditions are given as

$$194 \quad \mu_{1 \rightarrow 2}(1) = \exp(-\epsilon \mathcal{L}_r(t_1; \theta)), \mu_{N_t \rightarrow N_t-1}(1) = 0.5.$$

195 Finally, we obtain the marginal

$$196 \quad \mathbb{P}(X_i = 1) = b_i(1) = \mu_{i+1 \rightarrow i}(1) \cdot \mu_{i-1 \rightarrow i}(1) = \exp(-\epsilon \sum_{j=1}^{i-1} \mathcal{L}_r(t_j; \theta)),$$

199 which is equivalent to the definition of causal weights.

201 4 MESSAGE-PASSING WEIGHTS FOR TRAINING PINNS FOR
 202 TIME-DEPENDENT PDES
 203

204 The remaining problem is how to build the weights adapting the connection in Section 3. In fact, the
 205 normalizing process of BP algorithm was not displayed in Section 3, since they are already naturally
 206 normalized in L1 sense. Now, we shall define the weights for training PINNs, namely *Message-*
 207 *Passing weights (MP weights)*, motivated by Section 3, which resemble the message-passing process
 208 of BP algorithm without the normalization process. For details, we first consider the domain $\Omega =$
 209 $[0, T] \times \prod_{k=1}^d I_k \subset \mathbb{R}^{d+1}$ whose boundary consist of the Cartesian products of closed intervals.
 210 Next, we define the undirected graph $G = (V, E)$ whose nodes are the set of uniformly spaced
 211 collocation points of Ω with a gap size of δ . Then we can define G as follows:

$$212 \quad V = \{(t_j, x_{i_1}, \dots, x_{i_d}) : j = 1, \dots, N_t, i_k = 1, \dots, N_k, k = 1, \dots, d\}$$

$$214 \quad E = \{(v, w) \in V^2 : \|v, w\|_{L^1} = \delta\} \setminus (\{0\} \times \prod_{k=1}^d I_k)^2$$

where $\|\cdot\|_{L^1}$ is an L1 norm. Then Motivated from the Eq. 4 and the connection in 3, we can define the message-passing process on this graph, with the customized messages $\mu_{y \rightarrow x}^{(\tau)}(\theta) \in [0, 1]$ calculated for internal iteration steps $\tau = 1, \dots, D$ in each training loop, as:

$$\mu_{y \rightarrow x}^{(\tau+1)}(\theta) = \exp(-\epsilon \mathcal{N}[u_\theta](y)^2) \prod_{(y,z) \in E, z \neq x} \mu_{z \rightarrow y}^{(\tau)}(\theta), \quad \forall (x, y) \in E.$$

Especially, for two-dimensional time dependent case with $\Omega = [0, T] \times I$, the messages are defined as:

$$\begin{aligned} \mu_{(t,x) \rightarrow (t-\delta,x)}^{(\tau)}(\theta) &= 1, \\ \mu_{(t,x) \rightarrow (t+\delta,x)}^{(\tau+1)}(\theta) &= \exp(-\epsilon \mathcal{N}[u_\theta](t, x)^2) \mu_{(t-\delta,x) \rightarrow (t,x)}^{(\tau)}(\theta) \mu_{(t,x-\delta) \rightarrow (t,x)}^{(\tau)}(\theta) \mu_{(t,x+\delta) \rightarrow (t,x)}^{(\tau)}(\theta), \\ \mu_{(t,x) \rightarrow (t,x-\delta)}^{(\tau+1)}(\theta) &= \exp(-\epsilon \mathcal{N}[u_\theta](t, x)^2) \mu_{(t-\delta,x) \rightarrow (t,x)}^{(\tau)}(\theta) \mu_{(t,x-\delta) \rightarrow (t,x)}^{(\tau)}(\theta), \\ \mu_{(t,x) \rightarrow (t,x+\delta)}^{(\tau+1)}(\theta) &= \exp(-\epsilon \mathcal{N}[u_\theta](t, x)^2) \mu_{(t-\delta,x) \rightarrow (t,x)}^{(\tau)}(\theta) \mu_{(t,x+\delta) \rightarrow (t,x)}^{(\tau)}(\theta) \end{aligned}$$

whenever $t - \delta, x - \delta$ and $x + \delta$ are valid coordinates of points in V with fixed conditions

$$\mu_{v \rightarrow w}^{(\tau)}(\theta) = 1, \quad \forall v \in V \cap \partial\Omega, \forall w \in V \setminus \Omega$$

for each internal iteration $\tau = 1, \dots, D$ and

$$\mu_{v \rightarrow w}^{(1)}(\theta) = 1, \quad \forall v, w \in V.$$

Finally, we define the MP-weights

$$w^{\text{MP}}(x, \theta) := \prod_{(x,y) \in E} \mu_{y \rightarrow x}^{(D)}(\theta), \quad \forall x \in V$$

which is used to define the new residual loss $\mathcal{L}_r^{\text{MP}}(\theta)$ in Eq. 1 as

$$\mathcal{L}_r^{\text{MP}}(\theta) := \sum_{x \in V} w^{\text{MP}}(x; \theta) |\mathcal{R}[u](x)|^2, \quad \forall x \in V.$$

To facilitate the better understanding, we visualized the evolution of MP weights during the learning process for viscous Burgers equation in the Figure 1.

In practice, the hyperparameter D was set to 20. Additionally, adjusting causality has a crucial impact on the training and validation of PINNs. If the causality is too small, the differences in the residuals have less effect; if it is too large, convergence of the weights during the training may not be achieved. To measure the efficiency and effectiveness of loss balancing ability simultaneously, we applied the following process for determining causality: the causality parameter ϵ for each Causal PINN and MP-PINN was initially set to a sufficiently large value (practically set to 1000) and decreased exponentially by a factor of 1/10. This process was stopped and the value of ϵ was determined when the weights fully converged to 1, i.e. $\|w - 1\| < \delta = 0.1$, within the training epochs.

5 EXPERIMENTS

To demonstrate the performance of MP-weights, we aggregate the various time-dependent PDE examples as a benchmark from (Krishnapriyan et al., 2021; Wang et al., 2022a; Son et al., 2023), which are Convection, Allen-Cahn, viscous Burgers, and Klein-Gordon equation. Also, to verify the effectiveness of MP-PINN, we compared it with PINN, Causal PINN and AL-PINN that how MP-PINN overcome these methods and analyze the superiority over other adaptive weighting algorithms. The results are displayed in Table 1. Notably, We aim to show that the superiority of MP-PINN is due to its causal inference capabilities and present the following two experimental settings. First, each PDE has Dirichlet boundary conditions. This is because, from the perspective of supervised learning in PINNs, the initial and boundary conditions serve as true labels, which transmit information to the interior collocation points. Second, we propose early stopping based on loss without any learning scheduler during optimization. Since adaptive weighting algorithms adjust weights according to the residual magnitudes from the governing equation, initial, and boundary conditions, they can effectively guide the learning process. This suggests that these algorithms have an inherent ability to self-regulate learning. To directly compare this capability, we performed full-batch training using the Adam optimizer for 300,000 epochs. Finally, we used uniformly spaced collocation points, as well as initial and boundary points, for the training and test datasets.

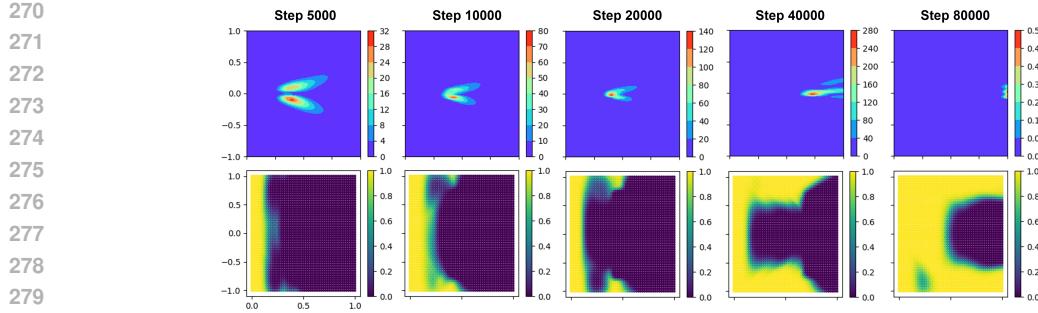


Figure 1: The process of increasing MP weights for viscous Burgers equation displayed with squared residuals (First row) and MP weight value (Second row) for each collocation points.

Table 1: Comparison of Relative L2 errors.

Experiments	PINNs	Causal PINNs	AL-PINNs	MP-PINNs
Convection	7.91e-03	2.34e-03	7.93e-03	2.51e-03
Allen-Cahn	5.24e-01	6.85e-02	6.45e-01	4.34e-02
viscous Burgers	3.18e-01	1.47e-02	4.29e-01	1.16e-02
Klein-Gordon	4.18e-03	2.43e-02	3.28e-03	1.26e-03

5.1 CONVECTION EQUATION

We first consider the Convection equation given as

$$\begin{aligned} \partial_t u + \beta \partial_x u &= 0, & \text{for } (t, x) \in [0, 1] \times [0, 2\pi], \\ u(0, x) &= \sin(x), & \text{for } x \in [0, 2\pi], \\ u(t, x) &= \sin(-\beta t), & \text{for } (t, x) \in [0, 1] \times \{0, 2\pi\} \end{aligned}$$

with $\beta = 30$. Figure 2 illustrates the absolute error between the true solution and the approximated solutions of each PINN. Both Causal PINN and MP-PINN achieved significantly better approximations compared to other PINNs. This demonstrates that MP-PINN has also successfully learned the time dependency in this problem.

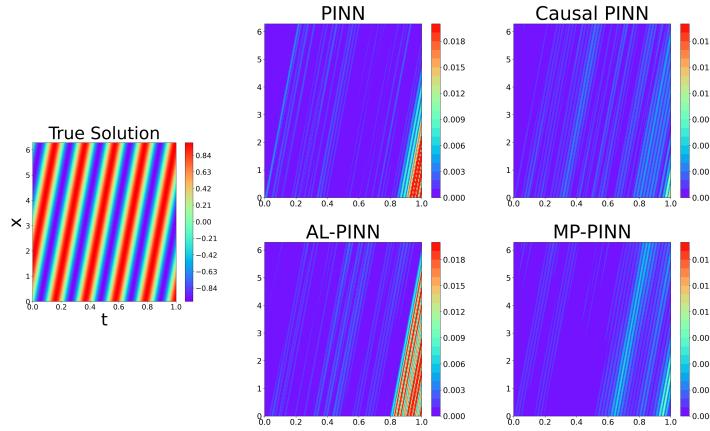


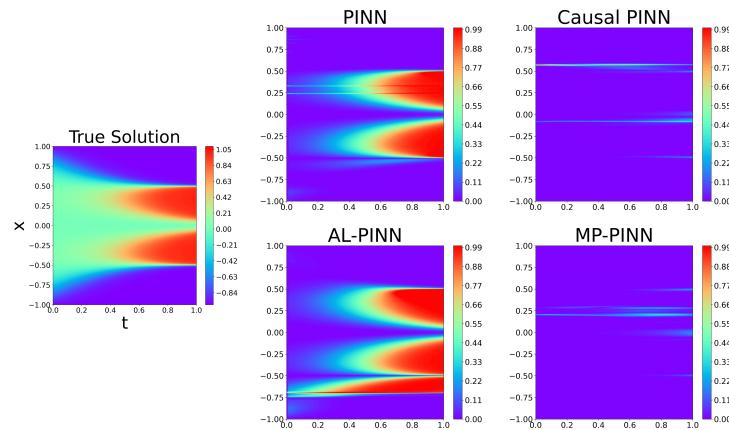
Figure 2: The true solution for Convection equation and comparison of absolute error map $|u_\theta - u|$ for each PINNs. In the error map, all values exceeding 0.02 were clipped to a constant.

324 **5.2 ALLEN-CAHN EQUATION**
 325

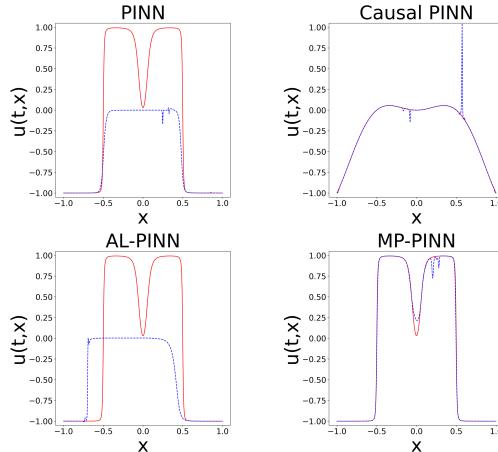
326 Next, we consider the Allen-Cahn equation given as
 327

$$\begin{aligned} \partial_t u + \beta \partial_x^2 u + 5u(u^2 - 1) &= 0, & \text{for } (t, x) \in [0, 1] \times [-1, 1], \\ u(0, x) &= x^2 \cos(\pi x), & \text{for } x \in [-1, 1], \\ u(t, x) &= -1, & \text{for } (t, x) \in [0, 1] \times \{-1, 1\} \end{aligned}$$

331 with $\beta = 10^{-4}$. Figure 3 illustrates that both Causal PINN and MP-PINN outperform other PINNs.
 332 However, this example demonstrates that spatial causality is also important for training PDE. Figure
 333 4 shows the cross-section of the Allen-Cahn equation where the mean absolute error is the largest
 334 for each weight across all time grids. As demonstrated, while Causal PINN fails to preserve spatial
 335 causality, MP-PINN maintains it even in the worst-case scenario. This highlights that MP-PINN
 336 respects the spatial causality of Allen-Cahn equation.
 337



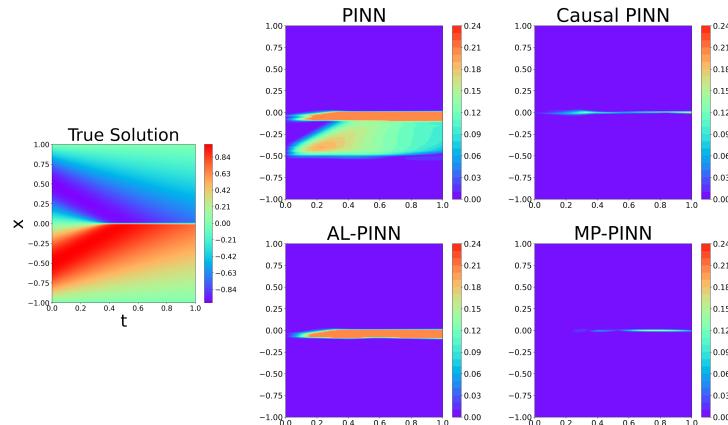
353 Figure 3: The true solution for Allen-Cahn equation and comparison of absolute error map $|u_\theta - u|$
 354 for each PINNs. In the error map, all values exceeding 1 were clipped to a constant.
 355



373 Figure 4: The comparison between the true solution (red dashed line) and the approximated solution
 374 (blue dotted line) from each PINN for the Allen-Cahn equation is shown at the time where the mean
 375 absolute error of the cross-section is the largest. The selected time points are $t = 0.0$ for Causal
 376 PINN and $t = 1.0$ for the others.
 377

378 5.3 VISCOUS BURGERS EQUATION
379380 To emphasize that MP-PINN effectively respects spatial causality, we considered the following vis-
381 cous Burgers equation:

382
$$\begin{aligned} \partial_t u + u \partial_x u - \nu \partial_x^2 u &= 0, & \text{for } (t, x) \in [0, 1] \times [-1, 1], \\ 383 u(0, x) &= -\sin(\pi x), & \text{for } x \in [-1, 1], \\ 384 u(t, x) &= 1, & \text{for } (t, x) \in [0, 1] \times \{-1, 1\} \end{aligned}$$

386 where $\nu = 0.01/\pi$. As demonstrated by the true solution in Figure 5, the viscous Burgers equation
387 exhibits very sharp spatial variations along the horizontal line $x = 0$. While Causal PINN struggles
388 to capture these changes, MP-PINN effectively smooths the steep spatial errors.
389405 Figure 5: The true solution for viscous Burgers equation and comparison of absolute error map
406 $|u_\theta - u|$ for each PINNs. In the error map, all values exceeding 0.2 were clipped to a constant.
407408 5.4 KLEIN-GORDON EQUATION
409410 Finally, we will investigate whether the MP-PINN effectively learns in the case where the second
411 derivative, rather than the first derivative of time, is provided in the given equation. For this, We
412 consider the Klein-Gordon Equation given as
413

414
$$\begin{aligned} \partial_t^2 u - \partial_x^2 u + u^3 &= f(t, x), & \text{for } (t, x) \in [0, 1] \times [0, 1], \\ 415 u(0, x) &= g_1(x), & \text{for } x \in [0, 1], \\ 416 \partial_t u(0, x) &= g_2(x), & \text{for } x \in [0, 1], \\ 417 u(t, x) &= h(t, x), & \text{for } (t, x) \in [0, 1] \times \{0, 1\}. \end{aligned}$$

418 with the unknown f, g_1, g_2, h are derived from the pre-given solution
419

420
$$u(t, x) = x \cos(5\pi t) + (tx)^3.$$

421

422 Figure 6 illustrated that all benchmark datasets, including Causal PINN, fail to capture the signifi-
423 cant information transmitted from the upper boundary $x = 1$, resulting in large errors in the central
424 diagonal region. In contrast, MP-PINN successfully reduces these errors, providing evidence of
425 respecting spatial causality.426 6 CONCLUSIONS
427428 In this study, we have developed a novel method, MP weights which respect the causality between
429 the training of each collocation points. This was achieved by considering not only time depen-
430 dency but also spatial causality, leading to superior performance respect to other adaptive weighting
431 algorithms and the ability to handle a wider variety of PDE types.

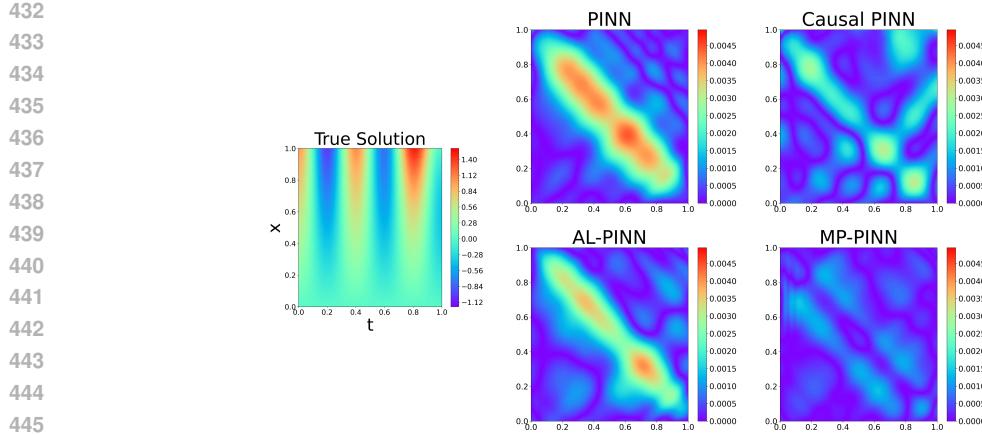


Figure 6: The true solution for Klein-Gordon equation and comparison of absolute error map $|u_\theta - u|$ for each PINNs.

A limitation of our research is the increase in computational cost for weight calculations and running time. Additionally, further theoretical analysis is required for the hyperparameter D , which represents the number of message-passing iterations. Specifically, as the number of collocation points increases, D inevitably needs to increase as well, requiring proper adjustment.

While this paper concludes with just a modification of the BP algorithm in the context of MP-PINN, in future research, we plan to investigate whether this can be related to actual causal inference. Additionally, we expect that more experiments for general types of PDEs can be conducted.

7 REPRODUCIBILITY STATEMENT

We provide the detailed experimental setup for each PDEs and PINNs in Appendix A.

REFERENCES

- Rui Chang, Jonathan R Karr, and Eric E Schadt. Causal inference in biology networks with integrated belief propagation. In *Pacific Symposium on Biocomputing Co-Chairs*, pp. 359–370. World Scientific, 2014.
- Arka Daw, Jie Bu, Sifan Wang, Paris Perdikaris, and Anuj Karpatne. Mitigating propagation failures in physics-informed neural networks using retain-resample-release (r3) sampling. *arXiv preprint arXiv:2207.02338*, 2022.
- Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pp. 1263–1272. PMLR, 2017.
- Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Message passing neural networks. *Machine learning meets quantum physics*, pp. 199–214, 2020.
- George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.
- Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W Mahoney. Characterizing possible failure modes in physics-informed neural networks. *Advances in Neural Information Processing Systems*, 34:26548–26560, 2021.
- Levi McClenny and Ulisses Braga-Neto. Self-adaptive physics-informed neural networks using a soft attention mechanism. *arXiv preprint arXiv:2009.04544*, 2020.

- 486 Michael Penwarden, Ameya D Jagtap, Shandian Zhe, George Em Karniadakis, and Robert M Kirby.
487 A unified scalable framework for causal sweeping strategies for physics-informed neural networks
488 (pinns) and their temporal decompositions. *Journal of Computational Physics*, 493:112464, 2023.
489
- 490 Rodrigo B Platte and Lloyd N Trefethen. Chebfun: a new kind of numerical computing. In *Progress
491 in industrial mathematics at ECMI 2008*, pp. 69–87. Springer, 2010.
- 492 Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A
493 deep learning framework for solving forward and inverse problems involving nonlinear partial
494 differential equations. *Journal of Computational physics*, 378:686–707, 2019.
- 495
- 496 Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini.
497 The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.
- 498
- 499 Hwijae Son, Sung Woong Cho, and Hyung Ju Hwang. Enhanced physics-informed neural networks
500 with augmented lagrangian relaxation method (al-pinns). *Neurocomputing*, 548:126424, 2023.
- 501
- 502 Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient pathologies
503 in physics-informed neural networks. *arXiv preprint arXiv:2001.04536*, 2020.
- 504
- 505 Sifan Wang, Shyam Sankaran, and Paris Perdikaris. Respecting causality is all you need for training
506 physics-informed neural networks. *arXiv preprint arXiv:2203.07404*, 2022a.
- 507
- 508 Sifan Wang, Xinling Yu, and Paris Perdikaris. When and why pinns fail to train: A neural tangent
509 kernel perspective. *Journal of Computational Physics*, 449:110768, 2022b.
- 510
- 511
- 512
- 513
- 514
- 515
- 516
- 517
- 518
- 519
- 520
- 521
- 522
- 523
- 524
- 525
- 526
- 527
- 528
- 529
- 530
- 531
- 532
- 533
- 534
- 535
- 536
- 537
- 538
- 539

540 **A EXPERIMENTAL SETUP**
 541

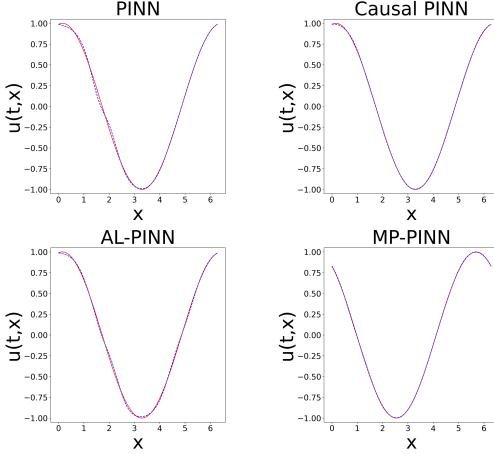
542 We adopt the true solution as the analytic solution for Convection, viscous Burgers and Klein-
 543 Gordon equation. For Allen-Cahn equation, we use chebfun package Platte & Trefethen (2010)
 544 to generate the numerical solution. The training settings for the benchmark PINNs, except for those
 545 specifically mentioned in the main text, are consistent with those in (Krishnapriyan et al., 2021;
 546 Wang et al., 2022a; Son et al., 2023), as detailed in Table 2.
 547

548 Table 2: Settings for training are displayed for each PINN (VN: PINN, CS: Causal PINN, AL: AL-
 549 PINN and MP: MP-PINN). For the network structure, MLP refers to a fully-connected network with
 550 hyperbolic tangent as activation functions, where all layers are initialized via Xavier initialization,
 551 and ResNet refers to an MLP with residual connections. The numeric values for Network Structure
 552 denote (hidden layer width) \times (hidden layer depth). The remaining settings are represented as fol-
 553 lows: learning rate η_θ for PINNs, initial and boundary coefficients λ_{ic} and λ_{bc} respectively, causality
 554 parameter ϵ for CS and MP, and learning rate η_λ for weight of AL.

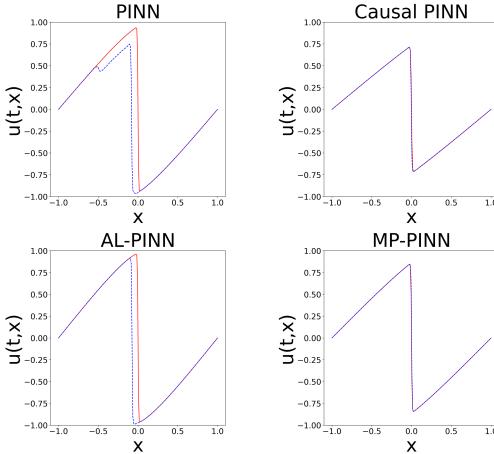
556 Experiment	557 Weight	558 Network 559 Structure	559 Train Grid / 560 Test Grid	560 η_θ	561 $\lambda_{ic}, \lambda_{bc}$	562 ϵ	563 η_λ
564 Convection	565 VN	566 CS	567 MLP, 568 50×4	569 $50 \times 50 /$ 570 201×512	571 10^{-3}	572 10^2	- -
		573 AL				- 1	
		574 MP				10^{-5}	-
		575 VN				- -	
576 Allen-Cahn	577 CS	578 MLP, 579 128×4	580 $50 \times 50 /$ 581 201×512	582 10^{-3}	583 10^2	584 10^2	-
		585 AL				- 1	
		586 MP				10^{-4}	-
		587 VN				- -	
588 viscous Burgers	589 CS	590 ResNet, 591 64×8	592 $50 \times 50 /$ 593 100×200	594 10^{-4}	595 1	596 10^2	-
		597 AL				- 10^{-3}	
		598 MP				10^{-4}	-
		599 VN				- -	
600 Klein-Gordon	601 CS	602 ResNet, 603 64×8	604 $50 \times 50 /$ 605 100×200	606 10^{-3}	607 5×10^2	608 10^{-2}	-
		609 AL				- 1	
		610 MP				10^{-8}	-

594 **B ADDITIONAL COMPARISON OF RESPECTING SPATIAL CAUSALITY**
 595

596 To clarify the results, we provide figures Figure 7, 8 and 9, comparing the cross-sectional error for
 597 other equations except Allen-Cahn.



615 Figure 7: The comparison between the true solution (red dashed line) and the approximated solution
 616 (blue dotted line) from each PINN for the Convection equation is shown at the time where the mean
 617 absolute error of the cross-section is largest. The selected time points are $t = 0.98$ for MP-PINN
 618 and $t = 1.0$ for the others.



636 Figure 8: The comparison between the true solution (red dashed line) and the approximated solution
 637 (blue dotted line) from each PINN for the viscous Burgers equation is shown at the time where the
 638 mean absolute error of the cross-section is largest. The selected time points are $t = 0.62$, $t = 1.0$,
 639 $t = 0.57$ and $t = 0.77$ for PINN, Causal PINN, AL-PINN and MP-PINN, respectively

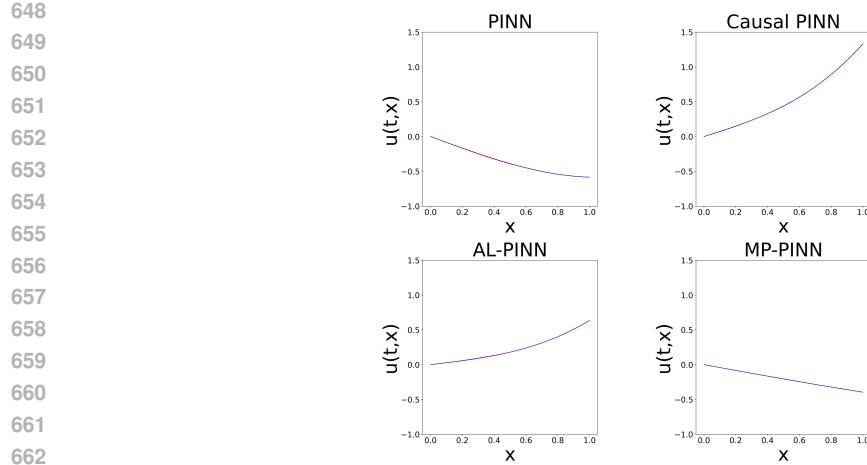


Figure 9: The comparison between the true solution (red dashed line) and the approximated solution (blue dotted line) from each PINN for the Klein-Gordon equation is shown at the time where the mean absolute error of the cross-section is largest. The selected time points are $t = 0.64$, $t = 0.85$, $t = 0.72$ and $t = 0.27$ for PINN, Causal PINN, AL-PINN and MP-PINN, respectively

648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701