1 Supplementary Text

1.1 Residual-based adaptive refinement

The original PINN framework proposed by Raissi et al. (2019) has difficulty in approximating the solution of PDEs that have steep gradients. To overcome this challenge, Lu et al. (2021) proposed the residual-based adaptive refinement (RAR) algorithm, which distributes collocation points during the training in the locations where the residual of PDEs is large.

In this study, as in Lu et al. (2021), after 10000 iterations of the Adam optimizer, the residual was evaluated at randomly sampled 10^6 locations from the whole spatial and temporal domain (see Fig. S1 (**b**)). The collocation points were ordered according to the residual values, and the highest ten collocation points were added to the collocation points that were originally given. We iterated this procedure ten times before the L-BFGS-B algorithm was used to further minimize the loss function.

We tested the RAR algorithm for the forward modeling of the homogeneous soil (Sect. 3.1 in the main text), and the results are shown in Fig. S1. The RAR algorithm appeared to improve the performance of PINNs for the problem, but the effects were minor. Therefore, we did not use the algorithm for further analysis.

1.2 Learning rate annealing

Wang et al. (2021) proposed the adaptive learning rate (ALR) algorithm, where the weight parameters in the loss function λ_i are updated in the following way:

$$\lambda_i^{n+1} = (1-\alpha)\lambda_i^n + \alpha\hat{\lambda_i}^n \quad \text{for} \quad i = m, ic, D, F, \tag{1}$$

where

$$\hat{\lambda_i}^n = \frac{\max_{\mathbf{W}^n}\{|\nabla_{\mathbf{W}^n}\mathcal{L}_r(\Theta^n)|\}}{\overline{|\nabla_{\mathbf{W}^n}\lambda_i^n\mathcal{L}_i(\Theta^n)|}}, \quad \text{for} \quad i = m, ic, D, F, \tag{2}$$

where the bar represents the mean of the values below the bar; Θ^n is the neural network parameters including the weight matrices $\mathbf{W}^n$ at nth iteration of the algorithm. In the study, α was set to 0.1, and the algorithm was used to update λ_i every 10 iterations of the the Adam algorithm to balance the relative importance of each loss term .

The ALR algorithm was tested for the forward modeling for the homogeneous soil case in Sect. 3.1 of the main text. The three weight parameters λ_{ic}, λ_{ub}, and λ_{lb} for the initial, upper boundary, and lower boundary condition, respectively, were initially set to ten, while they were updated using the ALR algorithm during the training (see Fig. S2 (**b**)). Figure S2 demonstrated that the effectiveness of the ALR algorithm was not clear compared to the L-LAAF algorithm. Figure S2 (**c**) showed that the loss term for the residual $\mathcal{L}_r$ was not minimized as the L-LAAF algorithm (shown in Fig. 4 of the main text). Therefore, we only used the L-LAAF algorithm in the study for further analysis.

1.3 Finite difference method

A finite difference method was implemented on Matlab R2020b to solve the one-dimensional RRE to evaluate the performance of PINNs. To deal with the non-linear terms in the RRE, the modified Picard iteration was used (Celia et al., 1990). A constant spatial mesh size dz and time step dt were used. The internodal hydraulic conductivity K was computed from the geometric average of the adjacent nodes. The upper boundary condition given as a constant water flux was evaluated using a second-order one-sided finite difference approximation (LeVeque, 2007). Figure S3 shows that the numerical error ϵ^θ decreased with decreasing dt (Fig. S3 (**a**)) and dz (Fig. S3 (**b**)).

1.4 Effects of weight parameters in loss function for heterogeneous case

In Sect. 3.2.4 in the main text, we investigated the effects of weight parameters λ_i in the loss function for the heterogeneous case. Here, the effects on the loss terms are shown. Figure S4, S5, and S6 shows the effects on the loss terms for the upper layer, the lower layer, and the interface conditions, respectively.

2 Supplementary Figures

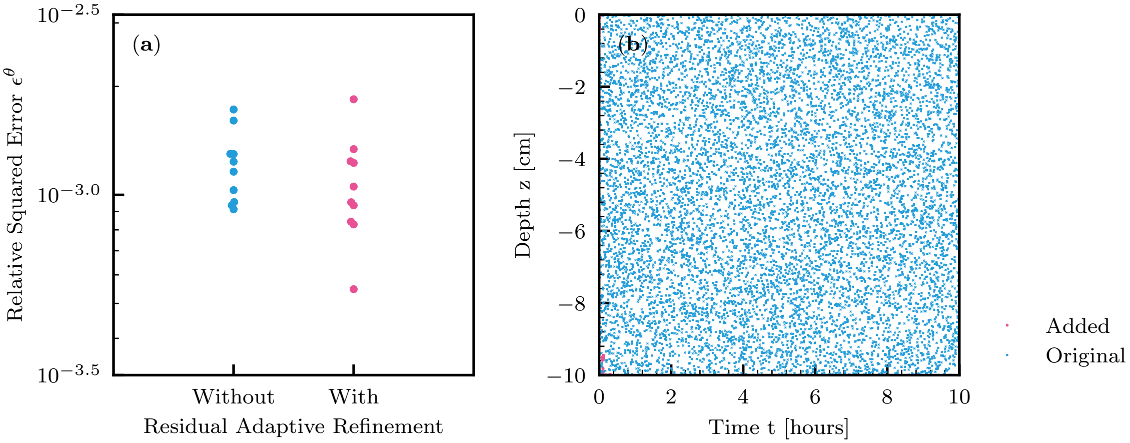


Figure S1. (**a**): The effects of the residual-based adaptive refinement algorithm on the performance of PINNs for the forward problem for the homogeneous soil. (**b**): The distribution of the original and added collocation points for the same problem.

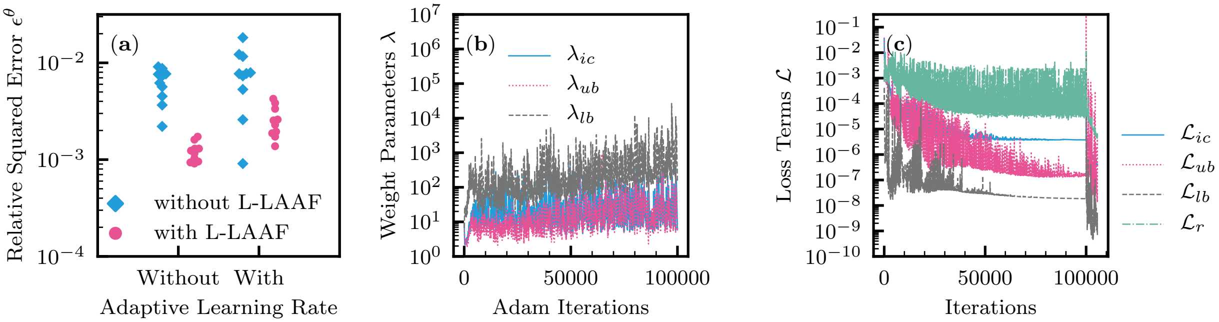


Figure S2. The effects of the adaptive learning rate (ALR) algorithm for the forward problem of the homogeneous soil case. (**a**): The relative squared error ϵ^θ for PINNs with and without the ALR and L-LAAF algorithms. (**b**): The evolution of the weight parameters in the loss function during the Adam algorithm. (**c**): The evolution of the loss terms during the training.

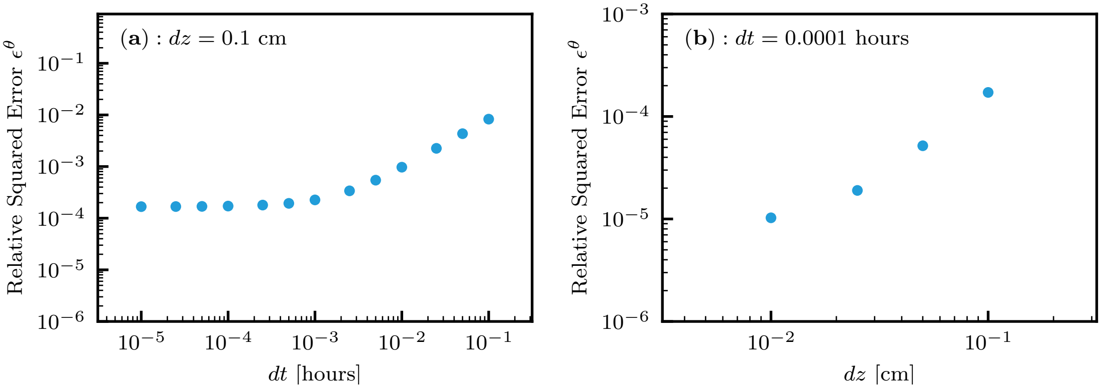


Figure S3. (**a**): The relative squared error with respect to volumetric water content ϵ^θ for the finite difference solution with varying time steps dt. The spatial mesh size dz was fixed to 0.1 cm. (**b**): The relative squared error with respect to volumetric water content ϵ^θ for varying spatial mesh size dz. The time step dt was fixed to 0.0001 h.

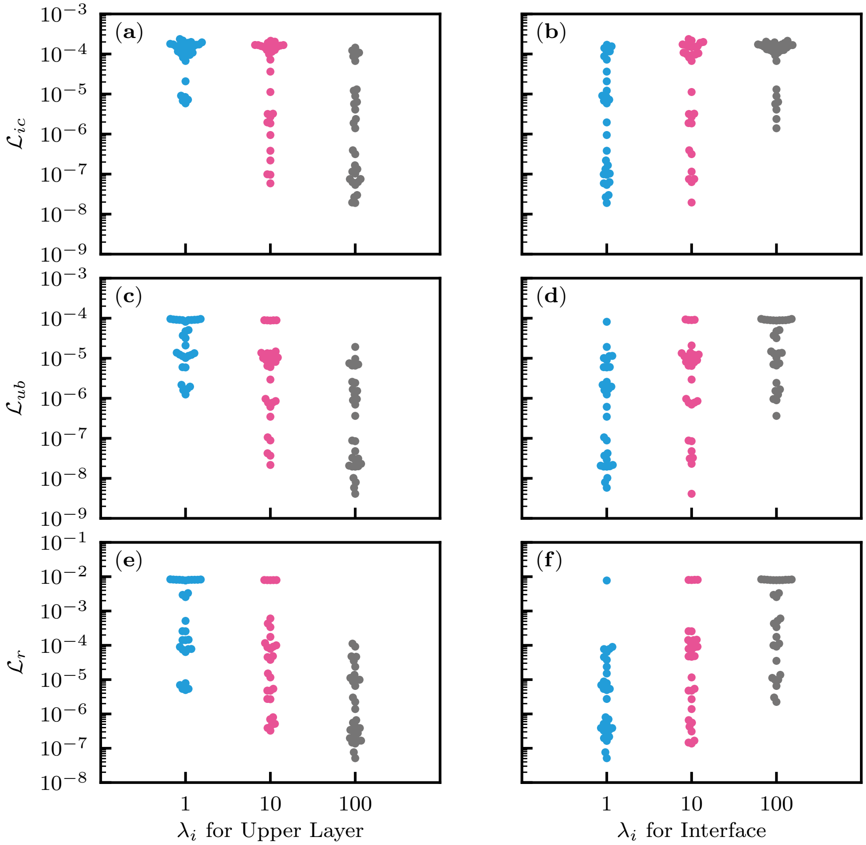


Figure S4. Heterogeneous soil. The effects of weight parameters λ_i in the loss function on the loss terms corresponding to the upper layer. The left and right columns correspond to the effects of λ_i for the upper layer and interface conditions, respectively. (**a**) and (**b**): Loss term for the initial condition $\mathcal{L}_{ic}$. (**c**) and (**d**): Loss term for the upper boundary condition $\mathcal{L}_{ub}$. (**e**) and (**f**): Loss term for the residual of the PDE $\mathcal{L}_{r}$.

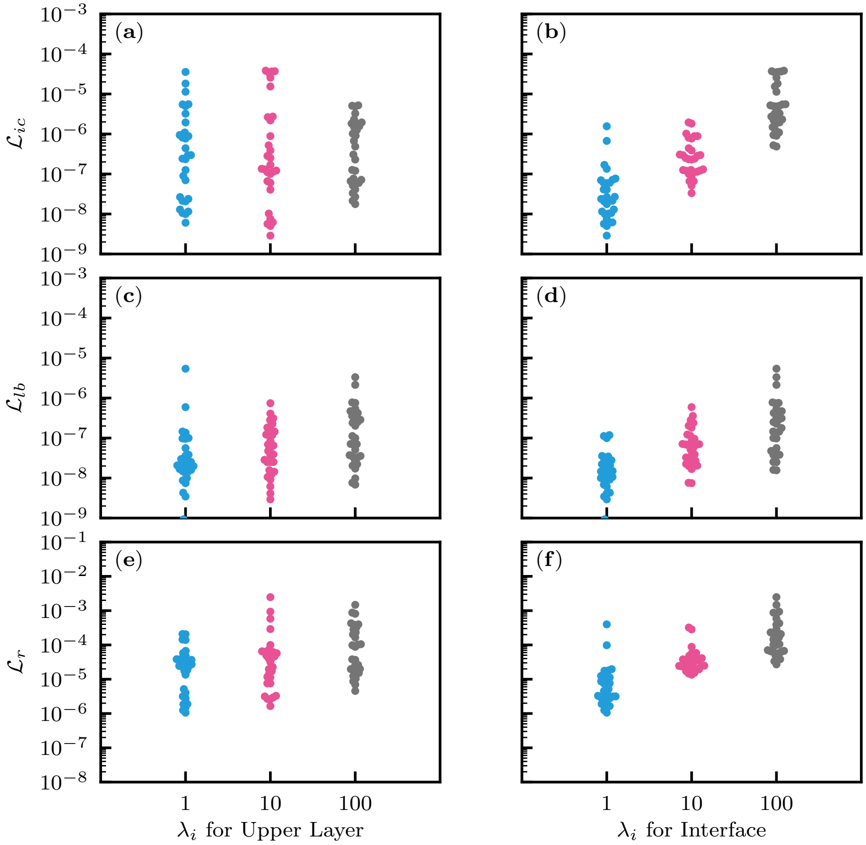


Figure S5. Heterogeneous soil. The effects of weight parameters λ_i in the loss function on the loss terms corresponding to the lower layer. The left and right columns correspond to the effects of λ_i for the upper layer and interface conditions, respectively. (**a**) and (**b**): Loss term for the initial condition $\mathcal{L}_{ic}$. (**c**) and (**d**): Loss term for the lower boundary condition $\mathcal{L}_{ub}$. (**e**) and (**f**): Loss term for the residual of the PDE $\mathcal{L}_r$.

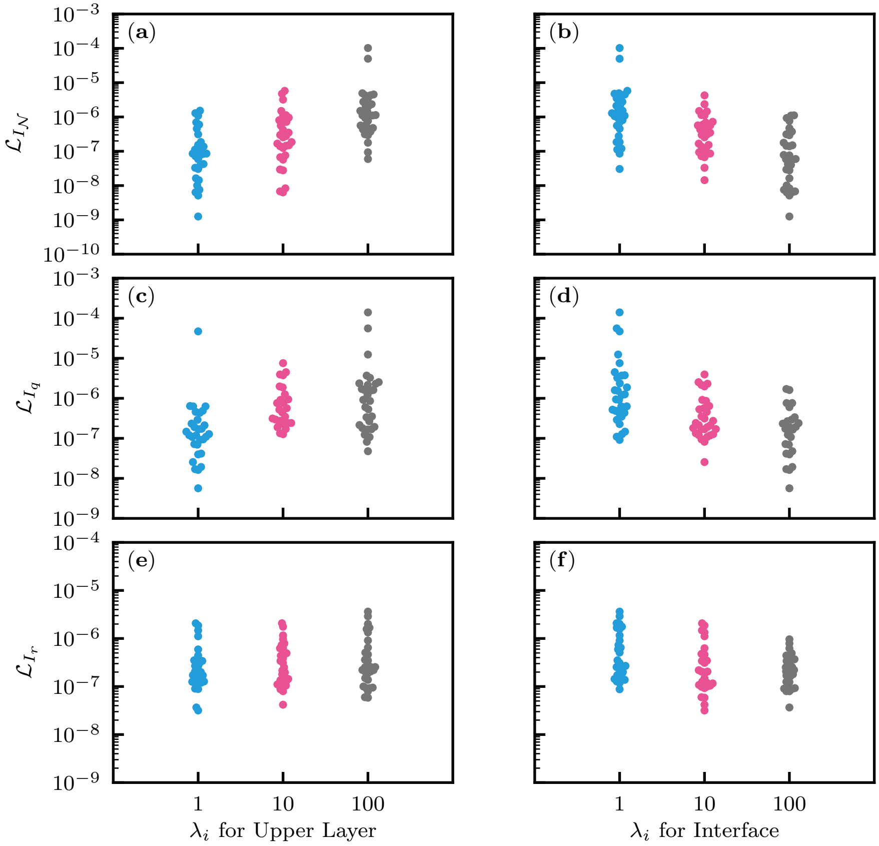


Figure S6. Heterogeneous soil. The effects of weight parameters λ_i in the loss function on the loss terms corresponding to the interface conditions. The left and right columns correspond to the effects of λ_i for the upper layer and interface conditions, respectively. (**a**) and (**b**): Loss term for the continuity in the neural network output $\mathcal{L}_{I_\mathcal{N}}$. (**c**) and (**d**): Loss term for the continuity in the water flux $\mathcal{L}_{I_q}$. (**e**) and (**f**): Loss term for the continuity in the residual of the PDE $\mathcal{L}_{I_r}$.

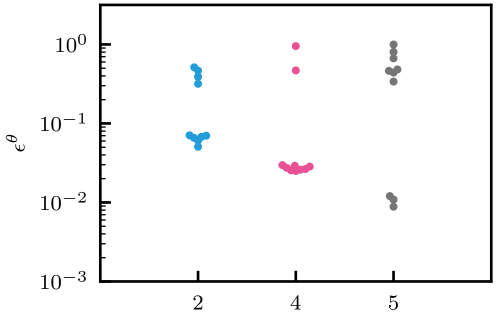
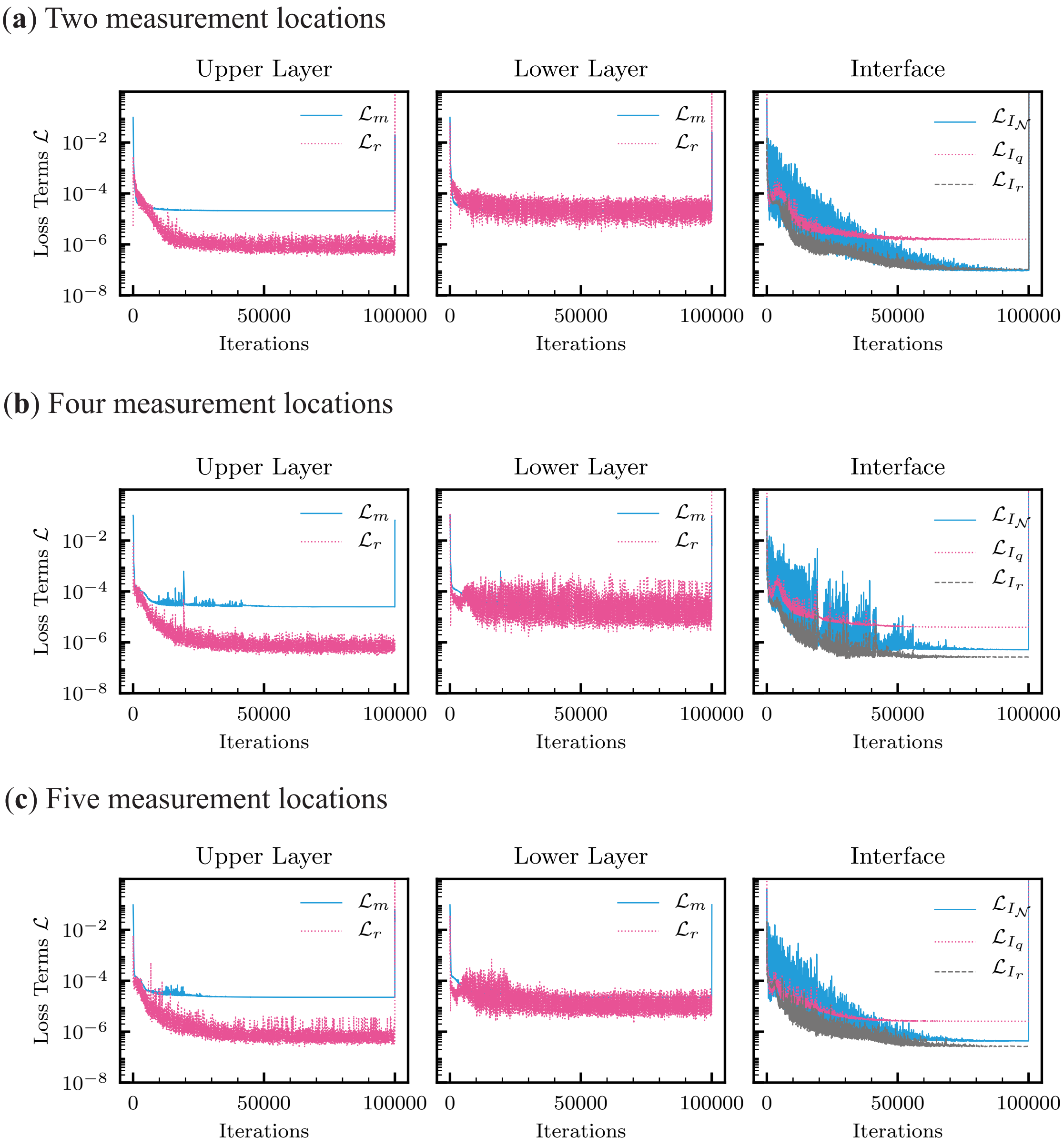


Figure S7. The relative squared error in terms of volumetric water content ϵ^θ for different numbers of measurement locations.



(**a**) Two measurement locations

(**b**) Four measurement locations

(**c**) Five measurement locations

Figure S8. Inverse modeling to estimate surface water flux from soil moisture measurements in a layered soil (upper layer: loam soil; lower layer: sandy loam soil). The evolution of loss terms for the upper layer (left column), lower layer (center column), and the interface conditions (right column) for different measurement locations z_m [cm]. (**a**): $z_m \in \{-5, -15\}$. (**b**): $z_m \in \{-3, -7, -13, -17\}$. (**c**): $z_m \in \{-1, -5, -9, -13, -17\}$.

References

Celia, M. A., Bouloutas, E. T., and Zarba, R. L.: A general mass-conservative numerical solution for the unsaturated flow equation, Water Resour. Res., 26, 1483–1496, https://doi.org/10.1029/WR026i007p01483, 1990.

LeVeque, R. J.: Finite difference methods for ordinary and partial differential equations, the Society for Industrial and Applied Mathematics, 2007.

Lu, L., Meng, X., Mao, Z., and Karniadakis, G. E.: DeepXDE: A deep learning library for solving differential equations, SIAM Review, 63, 208–228, https://doi.org/10.1137/19M1274067, 2021.

Raissi, M., Perdikaris, P., and Karniadakis, G. E.: Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, Journal of Comput. Phys., 378, 686–707, https://doi.org/10.1016/j.jcp.2018.10.045, 2019.

Wang, S., Teng, Y., and Perdikaris, P.: Understanding and mitigating gradient flow pathologies in physics-informed neural networks, SIAM J. Sci. Comput., 43, A3055–A3081, 2021.