IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 8 December 2023, accepted 9 January 2024, date of publication 16 January 2024, date of current version 29 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3354819

RESEARCH ARTICLE

Adaptive Uncertainty-Penalized Model Selection for Data-Driven PDE Discovery

PONGPISIT THANASUTIVES[1], TAKASHI MORITA[2,3], MASAYUKI NUMAO[2], (Member, IEEE), AND KEN-ICHI FUKUI[2], (Member, IEEE)

[1]Graduate School of Information Science and Technology, Osaka University, Suita, Osaka 565-0871, Japan
[2]SANKEN (The Institute of Scientific and Industrial Research), Osaka University, Ibaraki, Osaka 567-0047, Japan
[3]Academy of Emerging Sciences, Chubu University, Kasugai, Aichi 487-8501, Japan

Corresponding author: Pongpisit Thanasutives (thanasutives@ai.sanken.osaka-u.ac.jp)

ABSTRACT We propose a new parameter-adaptive uncertainty-penalized Bayesian information criterion (UBIC) to discover the stable governing partial differential equation (PDE) composed of a few important terms. Since the naive use of the BIC for model selection yields an overfitted PDE, the UBIC penalizes the found PDE not only by its complexity but also by its quantified uncertainty. Representing the PDE as the best subset of a few candidate terms, we use Bayesian regression to compute the coefficient of variation (CV) of the posterior PDE coefficients. The PDE uncertainty is then derived from the obtained CV. The UBIC follows the premise that the true PDE shows relatively lower uncertainty when compared with overfitted PDEs. Thus, the quantified uncertainty is an effective indicator for identifying the true PDE. We also introduce physics-informed neural network learning as a simulation-based approach to further validate the UBIC-selected PDE against the other potential PDE. Numerical results confirm the successful application of the UBIC for data-driven PDE discovery from noisy spatio-temporal data. Additionally, we reveal a positive effect of denoising the observed data on improving the trade-off between the BIC score and model complexity.

INDEX TERMS Bayesian regression, data-driven discovery, denoising, information criterion, model selection, partial differential equations, physics-informed neural networks, SINDy, uncertainty quantification.

I. INTRODUCTION

Data-driven discovery has emerged as an accurate approach for uncovering the governing partial differential equation (PDE) of a dynamical system without explicitly handling complex nonlinear relationships, offering greater flexibility than deriving physics from first principles (see AI Feynman [1] as a case study). Typically, sparse regression is leveraged to approximate a linear combination of candidate terms that balances between the capability to estimate a system state's temporal derivatives and the model complexity, hence so-called SINDy (sparse identification of nonlinear dynamics)-based approaches [2], which were successfully applied in aerodynamics [3], biology [4], [5], and epidemiology [6].

Regularized regression methods, designed to achieve the sparse identification were, for example, STRidge (sequential threshold ridge regression used in PDE-FIND) [7], LASSO (least absolute shrinkage and selection operator) [8] and SR3 (Sparse relaxed regularized regression) [9]. However, tuning a regularization hyperparameter(s) of these SINDy-based methods for model selection is challenging, because an improper hyperparameter may deliver an overfitted or underfitted model. Also, it is difficult to control the resulting model complexity. Thus, there is a risk of overlooking the true governing equation of a particular complexity.

The mixed-integer optimization (MIO) for best-subset selection [10] has been introduced to customize the desirable sparsity in the provably optimal MIO-SINDy [11]. Various best-subset solvers are utilized to collect potential PDEs, from which one is automatically selected as the underlying PDE by an algorithm [12]. Although impressive progress has been made in deliberate consideration of likely

The associate editor coordinating the review of this manuscript and approving it for publication was Alba Amato.

PDEs consecutively arranged in an order of increasing complexity, the important question remains: *How do we select the best model that reveals the true governing PDE form?*

Prior to model selection, we denoise the noisy observed data as we get a better chance of finding the true PDE form within a set of higher-quality potential PDEs. This step can be achieved by, for example, derivative computation using polynomials [7], spline-based models [13], and Robust PCA (principal component analysis) [14]. In this paper, we focus on using a denoising method, configured with a minimal set of hyperparameters that do not involve strictly assumed noise statistics (in contrast to a design of the Kalman filter for noise reduction), to yield a positive impact on the model selection step: the regularized K-SVD [15], a dictionary learning for computing sparse representations used to reconstruct denoised observed data. A performance comparison of different denoising methods is given in Appendix C.

In the model selection from a finite set of potential PDEs, Akaike information criterion (AIC) [16], [17], [18] and Bayesian information criterion (BIC) [19] are commonly adopted as metrics for evaluating point estimates of model parameters. However, when changing the number of nonzero terms in a linear model fitted on an overcomplete candidate library, the AIC and BIC values tend to decrease as the model complexity increases. Therefore, naively selecting the PDE that minimizes the information criteria, could lead to overfitted equations with unnecessary candidates [12], [20].

In this paper, we propose a new BIC-based metric that balances the accuracy of an approximated model not only by the model complexity but also by the quantified uncertainty to avoid overfitting. We go beyond point estimates of the model parameters or coefficients and instead put a posterior belief on them. Bayesian linear regression is modeled on each potential PDE to obtain the posterior distribution of the coefficients. The coefficients' mean and covariance are then computed, through the posterior samples or analytical methods (if available), to quantify the uncertainty using the coefficient of variation (CV) formula. Unlike the traditional BIC, our proposed uncertainty-penalized Bayesian information criterion (UBIC) adaptively exploits the uncertainty of the estimated coefficients to optimally select the parsimonious and stable governing PDE, which separates between overfitted and underfitted PDEs. Fig. 1 visualizes the primary steps of our approach to discover the underlying PDE starting from denoising data to model selection.

We suggest that the UBIC-selected PDE can be optionally validated using simulation-based model selection, measuring the BIC of the PDE simulated state solution for directly predicting the denoised observed data. We use a physics-informed neural network (PINN) [21] as a differentiable automatic PDE solver. PINN enables the flexible PDE-solving approach by incorporating physical laws as a part of its learning constraints. Physics-informed machine learning extends beyond solving PDEs, manifesting recent applications in learning nonlinear operators [22], [23]. Other than PINNs, frameworks based on Gaussian process regression [24], [25], [26] were introduced to solve a PDE or infer the PDE's parameters.

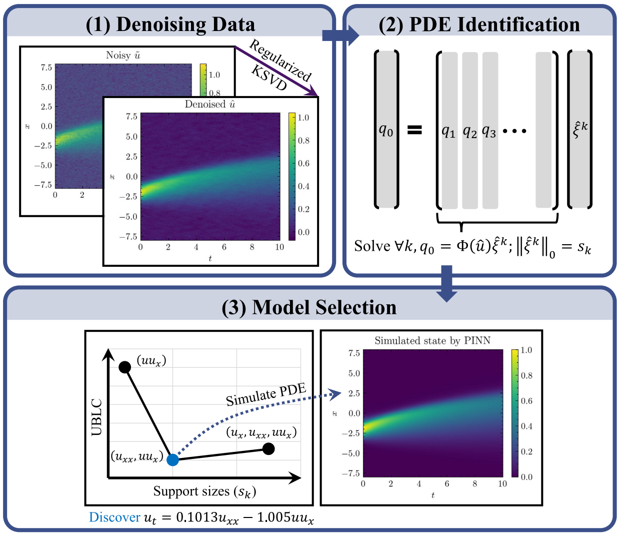


FIGURE 1. Schematic diagram of the Burgers' PDE discovery using the proposed UBIC for the model selection.

We list the main contributions in what follows.

- The UBIC uses the quantified uncertainty of each potential PDE to penalize the BIC, easing the identification of the parsimonious and stable governing PDE without heavy reliance on hyperparameter tuning. The quantified uncertainty is not exclusive to our approach but applicable to the existing methods that require model selection.
- We numerically exhibit the positive impact of denoising in terms of improving the trade-off given by the BIC.
- We explore the simulation-based model selection that evaluates the efficiency of the PINN-simulated PDE state solution in predicting the denoised observed data.

II. RELATED WORK

Pioneering sparse regression based PDE discovery methods, namely STRidge, LASSO, and SR3, relied on norm-based model selection techniques (i.e., minimizing regularized loss functions within a maximum number of iterations) to uncover the governing equation. However, the sensitivity to the choice of regularization hyperparameters, which varied from small to large values, was problematic. In prior research, an information criterion (IC) was used either for simulation-based model selection [17], [27] or for choosing among models that estimate the system state's time derivatives [12], [20]. The former type needs more computational resources, while each work of the latter type applies a subsequent selection algorithm to the PDEs pre-screened by the IC to ultimately find the best PDE form. Because the BIC decreases the precision per parameter (candidate) as the sample size increases [28], it should not be solely used for selecting the governing PDE.

Normally, regression frameworks (e.g., Gaussian processes and neural networks) combined with model selection [29], [30] are well suited for data-driven discovery tasks. The true PDE terms can emerge through various spaces (e.g., frequency spaces [31] or manifold's embedding spaces [29], [30], [32]) and candidate representations (e.g., polynomial interactions [2] or weak-form representations [33], [34]).

Bayesian PDE discovery methods, such as uncertainty-quantified SINDy (UQ-SINDy) [35] and threshold sparse Bayesian regression with error bars [36], were previously introduced, but the idea of uncertainty-penalized IC for PDE discovery remains unexplored. The UQ-SINDy employed sparsifying priors to induce nonzeros terms identified with their posterior inclusion probabilities generally once, leading to risks of missing some correct terms or including incorrect ones. The threshold sparse Bayesian regression also risked excluding small yet important PDE coefficients possibly due to an oversight when choosing the threshold.

Different from all the prior works, our UBIC is the first IC that integrates quantified uncertainty values of potential PDEs and adapts to noisy observed data to successfully identify the stable governing PDE without any PDE simulations.

III. METHODOLOGY

A. PROBLEM FORMULATION

Let us assume without loss of generality that the system state u in a two-dimensional (2D) spatio-temporal grid of spatially distributed physical systems satisfies

$$\partial_t u = \mathcal{N}(u, \partial_x u, \partial_x^2 u, \ldots; \xi). \tag{1}$$

We aim to discover $\mathcal{N}$, a linear or nonlinear operator involving spatial derivatives of the state variable u only. The parametric dependency ξ is a constant vector-valued coefficient. For convenience, we consider $\partial_x u \equiv u_x$ and other notations alike. Since our observed input $\tilde{u}$ may be disturbed with noise, or mathematically given as $\tilde{u}_{ij} = u(x_i, t_j) + z(x_i, t_j)$, we begin our PDE discovery approach by denoising on $\tilde{u}$. Noise $z(x_i, t_j) \sim \frac{\epsilon\sigma_u}{100}\mathcal{N}(0, 1)$ is drawn from standard Gaussian distribution, and scaled proportionally to ϵ% of the standard deviation (sd) σ_u calculated over the domain.

B. DENOISING DATA

Suppose the spatio-temporal grid is in a 2D space, we turn $\tilde{u}$ into a zero-mean array stacking flattened patches, regarded as signals $S_p(\tilde{u}) \in \mathbb{R}^{p^2 \times f}$; where p and f determine the patch size and the number of features. We seek the dictionary $D \in \mathbb{R}^{p^2 \times c}$, whose each column is denoted by d_j, along with corresponding sparse code A to approximate $S_p(\tilde{u})$ by its sparse representation $DA \approx S_p(\tilde{u})$. To achieve the approximation, we solve the ρ-regularized dictionary learning problem:

$$\begin{aligned} \min_{D,A} & \left\| S_p(\tilde{u}) - DA \right\|_F^2 + \rho \left\| A \right\|_F^2 \\ \text{subject to } & \left\| d_j \right\|_2 = 1, \quad j = 1, \ldots, c \\ & \| a_l \|_0 \leq L, \quad l = 1, \cdots, f. \end{aligned} \tag{2}$$

A couple of optimized D and A is obtain through regularized K-SVD training iterations. With the final fixed D, the ultimate sparse code A is then found using the orthogonal matching pursuit (OMP) algorithm with $\lfloor \frac{p^2}{10} \rfloor$ transforming sparsity to reconstruct the denoised observed data $\hat{u} = S_p^{-1}(DA)$ from the patches, via the inverse function S_p^{-1}. $\|\cdot\|_F$ denotes the Frobenius matrix norm. We define $\mathcal{G}_{\hat{u}}(x_i, t_j) = \hat{u}_{ij}$ as the denoised state function.

If we encounter 3D or 4D spatio-temporal data, the denoised $\hat{u}$ is instead achieved efficiently by applying 2D Savitzky-Golay filters [37], [38], which can be used with SVD (singular value decomposition).

C. PDE IDENTIFICATION

Best-subset regression is subsequently used to recover a sequence of potential parsimonious PDEs (with their corresponding coefficients) represented by best-subset solutions with a maximal bound on support sizes (i.e., the number of nonzero terms). For instance, a careful forward-backward elimination algorithm [39] can be implemented as a best-subset solver.

We presume an overcomplete library $\Phi(\hat{u})$ collecting candidate terms (with a maximum derivative order) of the denoised observed data. The library $\Phi(\hat{u})$ is supposed to embed the information on the initial and boundary conditions. According to the weak formulation [33], i-th numerical value of j-th candidate (column-wise) in $\Phi(\hat{u}) \in \mathbb{R}^{N_\Omega \times N_q}$ is given by integrating over a local spatio-temporal subdomain Ω_i, whose (rectangular) lengths are H_x and H_t.

$$\begin{aligned} \Phi(\hat{u}) &= [\cdots \quad q_j \quad \cdots], \quad j = 1, \ldots, N_q; \\ q_j^i &= \int_{\Omega_i} w\phi_j \, d\Omega, \quad i = 1, \ldots, N_\Omega. \end{aligned} \tag{3}$$

ϕ_j is regarded as a candidate function, for example, $\mathcal{G}_{\hat{u}}^2$ and $\partial_x^2 \mathcal{G}_{\hat{u}}$. N_Ω is the number of domain centers; $\forall i$, (x_i^c, t_i^c). The smooth weight, e.g., $w = (\underline{x}^2 - 1)^2(\underline{t}^2 - 1)^2$; where $\underline{x} = (x - x_i^c)/H_x, \underline{t} = (t - t_i^c)/H_t$ conditioned by $(\underline{x}, \underline{t}) \in [-1, 1]^2$, is a viable function for discovering the Burgers' PDE as it vanishes along the boundary $\partial\Omega_i$. Note that higher polynomial orders are possible. By integration by parts on Equation (3), numerical noisy derivative evaluation of ϕ_j is carried out on the noiseless w instead. We use the implementation provided in the PySINDy package [40], [41]. Remark that the noise-tolerant representation by the convolutional weak formulation (CWF) [34] could be leveraged at the library construction stage as well.

We attain an estimate $\hat{\xi}^k$ of the PDE coefficients with its support set, $\text{supp}(\hat{\xi}^k) = \{\hat{\xi}_j^k \mid \left|\hat{\xi}_j^k\right| > 0\}$ of a s_k support size (cardinality), by solving the best-subset selection problem:

$$\hat{\xi}^k = \underset{\xi^k}{\arg\min} \left\| q_0 - \sum_{j=1}^{N_q} q_j \xi_j^k \right\|_2^2, \text{ subject to } \left\| \xi^k \right\|_0 = s_k; \tag{4}$$

where $q_0^i = \int_{\Omega_i} w \partial_t \mathcal{G}_{\hat{u}}\, d\Omega$ and $q_0 \approx \sum_{j=1}^{N_q} q_j \hat{\xi}_j^k = \Phi(\hat{u})\hat{\xi}^k$. Best-subset solvers, we experiment with to yield potential PDEs for an increasing sequence of support sizes $(s_k)_{k=1}^{N_s}$; where $N_s \leq N_q$, are based on MIO, SOS-1-formulated (type-1 specially ordered sets) [42] MIO-SINDy, FROLS (forward regression with orthogonal least squares) [43], [44] and L0BnB (branch-and-bound framework for sparse regression) [45].

D. UNCERTAINTY-PENALIZED BAYESIAN INFORMATION CRITERION (UBIC) FOR ADAPTIVE MODEL SELECTION

We now find the best support size presented in Equation (4) within a given range. The base information criterion, on which we rely to penalize the maximized log-likelihood value of a regression model by its complexity, is the BIC:

$$\begin{aligned} \mathrm{BIC}(\hat{\xi}^k) &= -2\log L(\hat{\xi}^k) + \log(N_\Omega)s_k; \\ \log L(\hat{\xi}^k) &= -\frac{N_\Omega}{2}\log(\frac{2\pi}{N_\Omega}\left\| q_0 - \Phi(\hat{u})\hat{\xi}^k \right\|_2^2) - \frac{N_\Omega}{2}. \end{aligned} \tag{5}$$

L is the model likelihood function. Our motivating assumption is that the true governing PDE is reliable and thus parameterized by the stable vector-valued coefficient whose uncertainty is relatively lower (a good parsimony indicator) than those that characterize the other potential PDEs. Addressing the issue that the PDE with the lowest BIC is not necessarily the best PDE [12], [20], we define the UBIC by penalizing the BIC formula by tunable quantified uncertainty as follows:

$$\begin{aligned} \mathrm{UBIC}(\xi^k, \lambda_\mathrm{U}) &= \mathrm{BIC}(\xi_\mu^k) + \lambda_\mathrm{U} \log(N_\Omega)\mathrm{U}^k \\ &= -2\log L(\xi_\mu^k) + \log(N_\Omega)(s_k + \lambda_\mathrm{U}\mathrm{U}^k). \end{aligned} \tag{6}$$

U^k represents an estimated total uncertainty for ξ^k, scaled proportionally to $\log(N_\Omega)$, similar to the penalizing complexity in the BIC, for convenient unification. Like $L(\xi_\mu^k)$ and s_k, U^k is considered as an indicator of the PDE's parsimony. The data-dependent λ_U controlling influence of U^k on model selection is adaptively adjusted by Algorithm 1. A lower UBIC conveys a better-discovered PDE. A Bayesian linear regression probabilistic view is placed on ξ^k with Gaussian conjugate prior $\mathcal{N}(\xi^k \mid \xi_0^k, \mathrm{V}_0^k)$. Using Bayes rule for linear Gaussian systems [46], we derive the posterior as follows:

$$\begin{aligned} p(\xi^k \mid \Phi^k(\hat{u}), q_0, \sigma_q^2) &\sim \mathcal{N}(\xi^k \mid \xi_0^k, \mathrm{V}_0^k)\mathcal{N}(q_0 \mid \Phi^k(\hat{u})\xi^k, \sigma_q^2 \mathbf{I}_{N_\Omega}) \\ &= \mathcal{N}(\xi^k \mid \xi_\mu^k, \mathrm{V}^k); \\ \xi_\mu^k &= \mathrm{V}^k(\mathrm{V}_0^k)^{-1}\xi_0^k + \frac{1}{\sigma_q^2}\mathrm{V}^k\, \Phi^k(\hat{u})^T q_0, \\ \mathrm{V}^k &= \sigma_q^2(\sigma_q^2(\mathrm{V}_0^k)^{-1} + \Phi^k(\hat{u})^T \Phi^k(\hat{u}))^{-1}. \end{aligned} \tag{7}$$

Experimental results are produced with $\xi_0^k \in \mathbb{R}^{s_k}$, a vector containing nonzero terms in $\hat{\xi}^k$, and $\mathrm{V}_0^k = \mathbf{I}_{s_k}$ as an identity matrix of size s_k. Note that $\xi_0^k = \vec{0}$, reducing the posterior mean to ridge estimate, is also a feasible option. Columns of $\Phi^k(\hat{u})$ correspond to s_k effective candidates, which are used to calculate $\mathrm{BIC}(\xi_\mu^k)$. By maximum likelihood estimation (MLE) for the error variance, we set $\sigma_q^2 = \mathbb{E}[(q_0 - \Phi(\hat{u})\hat{\xi}^k)^2]$. Based on the obtained posterior, the uncertainty U^k is defined as follows:

$$\mathrm{U}^k = \frac{\mathrm{CV}^k}{\min_k \mathrm{CV}^k}; \; \mathrm{CV}^k = \frac{\left\| \mathrm{diag}(\mathrm{V}^k)^{\circ\frac{1}{2}} \right\|_1}{\left\| \xi_\mu^k \right\|_1} = \frac{\sum_{i=j}^{s_k} \sqrt{\mathrm{V}_{ij}^k}}{\left\| \xi_\mu^k \right\|_1}. \tag{8}$$

We compute CV^k (the relative standard deviation) of the covariance matrix V^k by taking an element-wise square root (the $\circ\frac{1}{2}$ exponent) on its diagonal vector (diag) and then the l_1-norm division by $\left\| \xi_\mu^k \right\|_1$. When all the true terms are included, the Bayesian linear model relies on them to approximate q_0, leaving the contribution of unnecessary terms on improving the approximate error diminished and uncertain with potentially high-variance coefficients. As $\sum_{i=j}^{s_k} \sqrt{\mathrm{V}_{ij}^k}$ sums the posterior standard deviation of every effective candidate, the more unnecessary candidates get included, the more the PDE risks becoming uncertain and overfitted and getting penalized more in Equation (6). Each CV^k is rescaled by the minimum $\min_k \mathrm{CV}^k$, resulting in U^k whose value is comparable to s_k.

Once every U^k is obtained, we converge the UBIC by Algorithm 1, iteratively decreasing λ_U from its maximum bound $\lambda_\mathrm{U}^{\max}$ derived to maintain the influence of the log-likelihood value (by not overly penalizing $-2\log L(\hat{\xi}^k)$ in Equation (6)) for all the discovered PDEs. We compute $\lambda_\mathrm{U}^{\max}$ based on the following constraint:

$$\begin{aligned} &\forall k \leq N_s,\; \log N_\Omega(s_k + \lambda_\mathrm{U}\mathrm{U}^k) \leq \left| -2\log \hat{L}(\xi_\mu^k) \right|; \; \lambda_\mathrm{U} \geq 0, \\ &\lambda_\mathrm{U}^{\max} = \max_k \frac{1}{\mathrm{U}^k}(\frac{2\left|\log \hat{L}(\xi_\mu^k)\right|}{\log N_\Omega} - s_k). \end{aligned} \tag{9}$$

Algorithm 1 finds a proper $\lambda_\mathrm{U} = 10^\lambda$ by reducing λ iteratively. We track the current and competitive optimal support sizes (s_{k^*}, s_{k^c}), and test the stopping condition at line 12, which essentially checks whether we have the increased complexity with unsatisfactory improvement (see τ), or the decreased complexity with already satisfying improvement. $\tau = \tau_0$ might be included in the stopping condition by choice. Also, τ_0 can be set adaptively, yet offering the same correct selection as the default value. Such an effective heuristic is $\tau_0 = P_{75}(S)$; $S = \{\tau_{k^1}^{k^2} \mid k^1, k^2 = \arg\min(r) \text{ s.t. } r = s_{k^2} - s_{k^1} > 0, \text{ and } \forall s_{k^0} < s_{k^1}, \mathrm{BIC}(\xi_\mu^{k^2}) < \mathrm{BIC}(\xi_\mu^{k^1}) < \mathrm{BIC}(\xi_\mu^{k^0})\}$, the 75th percentile of successive improvement factors respecting just BIC-decreasing models. If an overfitted model is detected by line 18, we retry with a stricter percentile of S, e.g.,

Algorithm 1 Find the Optimal Complexity s_{k^*} by Tuning λ_{U}

Input: $\Phi(\hat{u})$, q_0 and $\hat{\xi}^k$

Parameter: τ_0: Improvement threshold (default $= 0.02$) and N_δ: maximum number of iterations (default $= 3$)

Output: The optimal support size s_k^* and tuned UBIC's hyperparameter λ_{U}

```
Compute $\forall k \leq N_s$, $\mathrm{V}^k$, $\xi_\mu^k$ and $\mathrm{U}^k$,
with Gaussian prior $\mathcal{N}(\xi^k \mid \xi_0^k, \mathbf{I}_{s_k})$
Assign $\lambda \leftarrow \log_{10}\max(\lambda_{\mathrm{U}}^{\max}, 0)$ $\{-\infty$ if $\lambda_{\mathrm{U}}^{\max} \leq 0\}$
Assign $\delta \leftarrow \frac{\lambda}{N_\delta}$ and $\lambda^c \leftarrow \lambda - \delta$ {next trial value of $\lambda$}
Compute $\forall k$, $\mathcal{I}_k \leftarrow \mathrm{UBIC}(\xi^k, 10^\lambda)$ using $\xi_\mu^k$ and $\mathrm{U}^k$
Find $s_{k^*}$ where $k^* \leftarrow \arg\min_k \mathcal{I}_k$
while $\lambda^c \geq 0$ do
    Compute $\forall k$, $\mathcal{I}_k^c \leftarrow \mathrm{UBIC}(\xi^k, 10^{\lambda^c})$
    Find $s_{k^c}$ where $k^c \leftarrow \arg\min_k \mathcal{I}_k^c$
    Assign $\Delta s \leftarrow s_{k^c} - s_{k^*}$
    Assign $\Delta\mathrm{BIC} \leftarrow \mathrm{BIC}(\xi_\mu^{k^c}) - \mathrm{BIC}(\xi_\mu^{k^*})$
    Assign $\tau \leftarrow \tau_{k^*}^{k^c}$; $\tau_{k^*}^{k^c} = \left|\Delta\mathrm{BIC}/(\mathrm{BIC}(\xi_\mu^{k^*})\Delta s)\right|$
    if ($\Delta s > 0$ but ($\Delta\mathrm{BIC} > 0$ or $\tau < \tau_0$)) or
    ($\Delta s < 0$ but $\Delta\mathrm{BIC} > 0$ and $\tau > \tau_0$) then
        break{stopping condition detected}
    end if
    Assign $\lambda \leftarrow \lambda^c$ and $\lambda^c \leftarrow \lambda - \delta$
    Assign $\forall k$, $\mathcal{I}_k \leftarrow \mathcal{I}_k^c$ and $k^* \leftarrow k^c$ $\{s_{k^*} \leftarrow s_{k^c}\}$
end while
if $\left|\mathrm{BIC}(\xi_\mu^{k^*}) - \mathrm{BIC}(\xi_\mu^{k^*-1})\right| / \left|\mathrm{BIC}(\xi_\mu^{k^*-1})\right| < \tau_0$ then
    Consider an increased or decreased $\tau_0$ value to prevent
    overfitted or underfitted models, respectively
end if
return $s_{k^*}$, $\lambda_{\mathrm{U}} = 10^\lambda$ and $\forall k$, $\mathcal{I}_k$ {used in plotting}
```

$P_{80}(S)$. On the contrary, lessening τ_0 helps discern selecting a supposedly underfitted 1-support-size PDE. The cost of computing UBIC scores is controlled by limiting the maximum support size s_{N_s} because the best subsets for all the support sizes have to be prepared in advance.

E. SIMULATION-BASED MODEL SELECTION

Since Equation (4) optimizes for the regression model that fits $\Phi(\hat{u})$ to approximate q_0 (a weak form of u_t), the attained models do not offer the direct comparison metric to the PDE solution. However, they should give rise to the true PDE, thereby reducing the computational burden of simulating false governing PDEs.

Conducting an aid validation process, we solve the found PDEs most likely to be the true PDE based on obtained UBIC scores. Well-known numerical PDE solvers using spectral methods are available in Chebfun [47] and Dedalus [48]. The software can accurately solve canonical stiff PDEs given initial and boundary conditions. Nevertheless, when solving a PDE containing extraneous high-order derivatives, the simulated solution may explode over time. While a rigorous mathematical analysis is beneficial to address ill-posed PDEs, our focus is on providing a flexible treatment using neural networks.

1) PHYSICS-INFORMED NEURAL NETWORK (PINN) LEARNING

PINN learning is an alternative approach for solving PDEs. The learned solution not only satisfies a specified PDE but also fits observational data. The general principle is to learn the mapping function from spatio-temporal data $\mathcal{D} = \{(x_i^{\mathcal{D}}, t_j^{\mathcal{D}})\}$ to denoised s_k-support-size PDE solution: $\forall i, j : (x_i^{\mathcal{D}}, t_j^{\mathcal{D}}) \overset{f_{\Theta_k}}{\mapsto} \hat{u}_{ij}^{\mathcal{D}}$ by minimizing the physics-informed loss respecting the discovered function $\hat{\mathcal{N}}$:

$$\min_{\Theta_k, \hat{\xi}^k} \left(\left\| \mathcal{F}_{\Theta_k}^{\mathcal{D}} - \hat{u}^{\mathcal{D}} \right\|_F^2 + \left\| \partial_t \mathcal{F}_{\Theta_k}^{\mathcal{D}} - \hat{\mathcal{N}}(\mathcal{F}_{\Theta_k}^{\mathcal{D}}, \partial_x \mathcal{F}_{\Theta_k}^{\mathcal{D}}, \partial_x^2 \mathcal{F}_{\Theta_k}^{\mathcal{D}}, \ldots; \hat{\xi}^k) \right\|_F^2 \right). \tag{10}$$

We collect $(\mathcal{F}_{\Theta_k}^{\mathcal{D}})_{ij} = f_{\Theta_k}(x_i^{\mathcal{D}}, t_j^{\mathcal{D}})$. $\mathcal{D}_{\mathrm{Train}}$ is the train split containing subsampled discretized spatio-temporal points, on which the PINN is trained, and likewise $\mathcal{D}_{\mathrm{Val}}$ for hold-out validation dataset. Automatic differentiation is used to compute derivative terms, e.g., $\partial_t \mathcal{F}_{\Theta_k}^{\mathcal{D}}$ and $\partial_x \mathcal{F}_{\Theta_k}^{\mathcal{D}}$. The full-batch second-order L-BFGS [49] is used to optimize Θ_k (i.e., learning to obtain Θ_k^*) and $\hat{\xi}^k$. We initialize $\hat{\xi}^k$ as the solution of Equation (4).

2) PHYSICS-INFORMED MODEL SELECTION

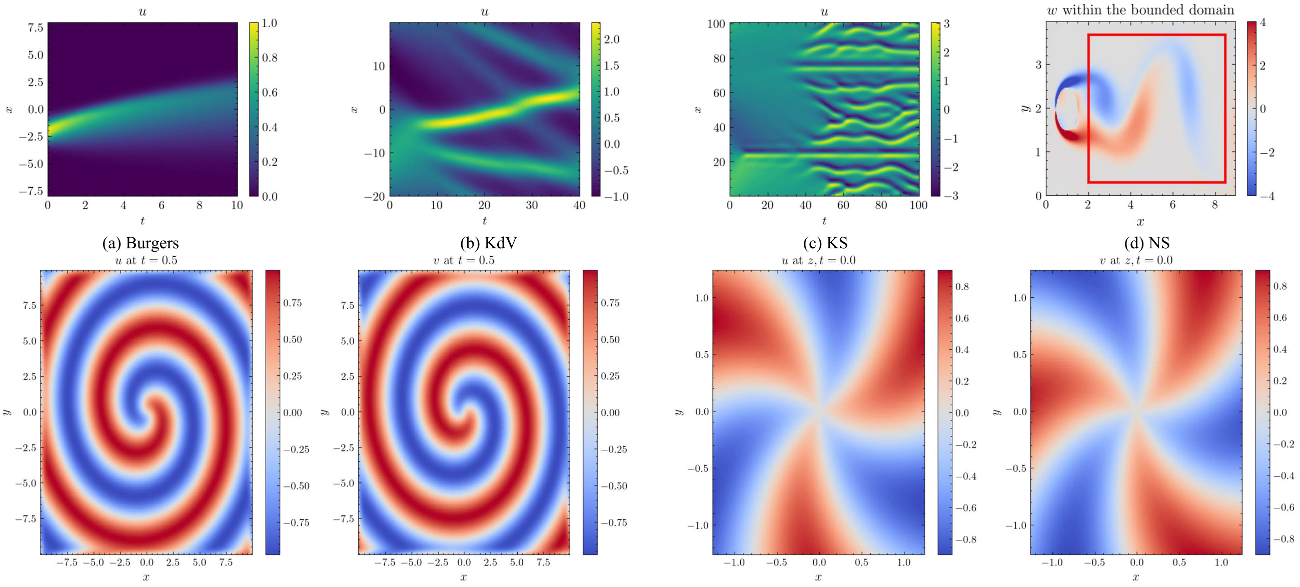
Deciding whether the s_k-support-size PDE is better or worse than the s_{k+1}-support-size PDE, we adjust the BIC in Equation (5) as the complexity penalization avoids choosing the overfitted PDE that also generates the PINN predicted solution close to $\hat{u}^{\mathcal{D}_{\mathrm{Val}}}$ on the validation split. Here, the state-level BIC reads

$$\mathrm{BIC}_{\Theta_k^*}^{\hat{u}}(\hat{\xi}^k) = |\mathcal{D}_{\mathrm{Val}}| \left(1 + \log\left(\frac{2\pi}{|\mathcal{D}_{\mathrm{Val}}|} \left\| \hat{u}^{\mathcal{D}_{\mathrm{Val}}} - \mathcal{F}_{\Theta_k^*}^{\mathcal{D}_{\mathrm{Val}}} \right\|_F^2 \right)\right) + \log(|\mathcal{D}_{\mathrm{Val}}|)(\left|\Theta_k^*\right| + \left\|\hat{\xi}^k\right\|_0); \tag{11}$$

where $\left|\Theta_k^*\right|$ tells the number of trainable parameters of the optimized neural network. The modification involves utilizing the validation data points in $\mathcal{D}_{\mathrm{Val}}$ for the comparison. If $\mathrm{BIC}_{\Theta_k^*}^{\hat{u}}(\hat{\xi}^k) < \mathrm{BIC}_{\Theta_{k+1}^*}^{\hat{u}}(\hat{\xi}^{k+1})$, it is advisable not to increase the support size to s_{k+1} under a fair circumstance where the PINN architecture and learning procedure are identical. We apply the vanilla paradigm for simplicity, though the PINN training could be improved, e.g., using multi-task learning techniques [50], since the network may overfit on $\mathcal{D}_{\mathrm{Train}}$, stuck in a physics-obeying local minimum [51]. In practice, the PINN network's size and number of epochs until convergence would be limited to fit one's computational resources.

TABLE 1. PDE dataset descriptions. The number of discretized spatial and temporal points are specified by the (N_x, N_y, N_z) and N_t. The intensity of ϵ%-sd Gaussian noise is listed in the rightmost column.

Dataset	PDE	N_x, N_y, N_z	N_t	ϵ
Burgers	$\partial_t u = 0.1\partial_x^2 u - u\partial_x u$	256 on $[-8, 8]$	101 on $[0, 10]$	30
KdV	$\partial_t u = -\partial_x^3 u - u\partial_x u$	512 on $[-20, 20]$	501 on $[0, 40]$	30
KS	$\partial_t u = -\partial_x^2 u - \partial_x^4 u - u\partial_x u$	1024 on $[0, 32\pi]$	251 on $[0, 100]$	30
NS	$w_t = 0.01(w_{xx} + w_{yy}) - uw_x - vw_y$	325×170 on $[2, 8.48] \times [0.3, 3.68]$	151 on $[0, 30]$	1
RD	$u_t = u - u^3 + v^3 - uv^2 + u^2 v + 0.1(u_{xx} + u_{yy})$ $v_t = v - u^3 - v^3 - uv^2 - u^2 v + 0.1(v_{xx} + v_{yy})$	256×256 on $[-10, 10] \times [-10, 10]$	201 on $[0, 10]$	10
GS	$u_t = 0.014 - 0.014u - uv^2 + 0.02(u_{xx} + u_{yy} + u_{zz})$ $v_t = -0.067v + uv^2 + 0.01(v_{xx} + v_{yy} + v_{zz})$	$128 \times 128 \times 128$ on $[-1.25, 1.25]^3$	100 on $[0, 10]$	0.1



(a) Burgers (b) KdV (c) KS (d) NS

(e) RD (f) GS

FIGURE 2. 2D visualization of the noiseless datasets.

IV. NUMERICAL RESULT AND DISCUSSION

The PDE dataset description is given in Table 1. We plot 2D visualization of the datasets we experimented with in Fig. 2. Experiments were run on a 2.6 GHz 6-Core Intel i7 CPU with 32 GB RAM. For reproducibility, the data and code are available at https://github.com/Pongpisit-Thanasutives/UBIC.

A. BURGERS' PDE

We tested the PDE solution with the initial condition: $u(x, 0) = e^{-(x+2)^2}$. To denoise $\tilde{u}$, we ran regularized KSVD with $\rho = 0.05$ on the stack $S_p(\tilde{u})$ created with square patches of size 8 × 8. For sparse encoding during the training, OMP was configured with one target sparsity.

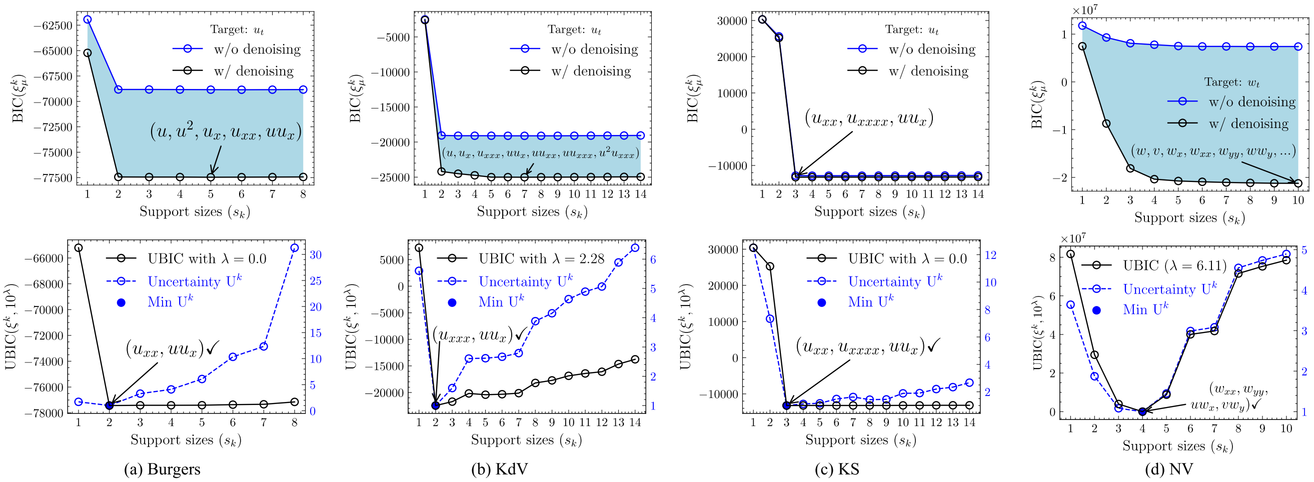
We gathered an overcomplete set of candidate functions, $\phi_j(\cdot) \in \{(\mathcal{G}_{\hat{u}}^{d_1} \partial_x^{d_2} \mathcal{G}_{\hat{u}})(\cdot) \mid d_1 + d_2 \geq 1; d_1, d_2 = 0, 1, 2\}$. For transforming to the integral weak forms $(\Phi(\hat{u}), q_0)$, we stick with 10000 domain centers throughout this paper. Exhaustive all subsets selection solved Equation (4), attaining $\hat{\xi}^k$ for every $k \leq N_q = 8$ (no constant term) in 0.29 secs (seconds).

Fig. 3(a) shows the BIC scores of the found PDEs, where the support sizes are arranged in increasing order. After the 2-support-size PDE, the improvement in BIC becomes stagnant. However, the model selection based on BIC does not choose the 2-support-size PDE as the optimal choice. This is because the BIC scores continue decreasing beyond the plateau, and it is the 5-support-size PDE that yields the lowest BIC score. We inspect that the log-likelihood dominates the BIC score, when the number of samples in the library N_Ω is large, causing the penalization by only the support size s_k (model complexity) not strong enough for identifying the true governing PDE.

To use the proposed UBIC, we quantify the uncertainty U^k for all the best subsets, as plotted in Fig. 3(a). These uncertainty values are incorporated to further penalize the obtained BIC scores, preferring the parsimonious PDE with the stable coefficient estimates. The PDE with a support size of 2 (2 effective candidates) exhibits the highest stability (inversely proportional to the PDE uncertainty). Algorithm 1 suggests the UBIC scores with $\lambda_{\mathrm{U}} = 10^0 = 1$. The UBIC-selected PDE aligns with the true Burgers' PDE form.

B. KORTEWEG-DE VRIES (KDV) PDE

We generated the two-soliton u with the initial condition $u(x, 0) = -\sin(\frac{\pi x}{20})$. We denoise using regularized KSVD with $\rho = 0.01$ on the stack $S_p(\tilde{u})$ created with 25×25 patches.



(a) Burgers (b) KdV (c) KS (d) NV

FIGURE 3. We plot the BIC, uncertainty U^k and UBIC with tuned $\lambda_U = 10^\lambda$ for the model selection in the Burgers, KdV, KS, and NS examples arranged from left to right. We use an arrow (→) to locate where the IC is minimized. "✓" indicates that the UBIC selects the true PDE form.

We set the OMP algorithm with one target sparsity during the training.

Next, the candidate functions $\phi_j(\cdot)$ were chosen from $\{(\mathcal{G}^{d_1}_{\hat{u}}\partial^{d_2}_x\mathcal{G}_{\hat{u}})(\cdot) \mid d_1 + d_2 \geq 1; d_1 = 0, 1, 2 \text{ and } d_2 = 0, 1, 2, 3, 4\}$ before building their weak forms and q_0. We exhaustively searched for all the optimal subsets respecting every support size, achieving $\forall k \leq N_q = 14$, $\hat{\xi}^k$ in 19.04 secs.

In Fig. 3(b), we observe that the true equation favored by the UBIC with tuned $\lambda_U = 10^{2.28}$ stands out, in accordance with the minimal uncertainty, from the other potential PDEs.

C. KURAMOTO-SIVASHINSKY (KS) PDE

Following the PDE-FIND paper [7], we experimented with the identical chaotic PDE generated with the initial condition: $u(x, 0) = \cos(\frac{x}{16})(1 + \sin(\frac{x}{16}))$. We used the same regularized KSVD settings as detailed in the KdV example.

Given that the set of candidate functions adopted in the KdV example was considered to build a weak-form library for recovering the KS PDE, we completed the same best-subset regression strategy in 32.22 secs.

As seen in Fig. 3(c), the lowest BIC and UBIC score with $\lambda_U = 1$ determine the true equation form.

D. NAVIER-STOKES (NS) PDE

We consider the explicit form of the NS equation given in the 3D spatio-temporal grid. As seen in Table 1, w denotes the vorticity. The components of the velocity field are denoted by u (x-component) and v (y-component), which are both treated as known terms to construct an overcomplete library. We generated the dataset according to the instructions given in the PDE-FIND paper and focused on the bounded spatial domain $(x, y) \in [2, 8.48] \times [0.3, 3.68]$ after the cylinder. We were left with $N_\Omega = 8342750$ data points for each variable—w, u, and v—to which we add 1%-sd noise after the subsampling.

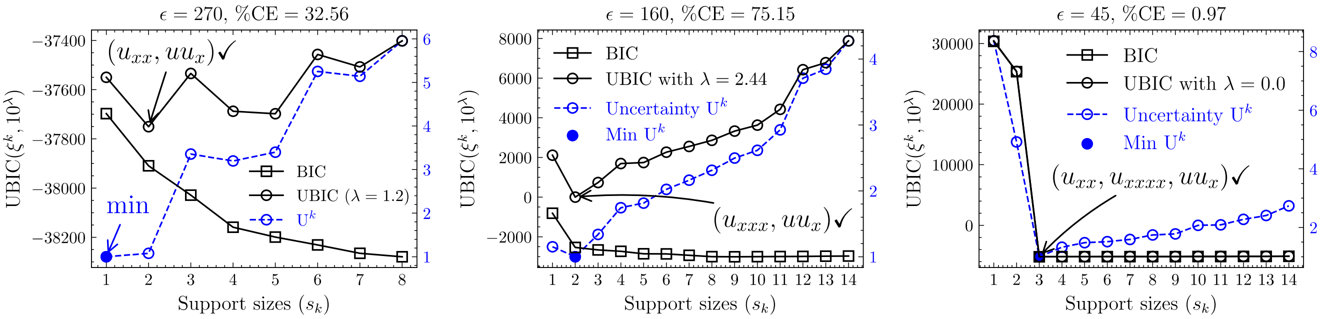
TABLE 2. The better is underlined (%CE) or on bold (R_{BIC}).

Dataset	w/o Denoising		w/ Denoising	
	%CE	R_{BIC}	%CE	R_{BIC}
Burgers	0.7900[1]	−6919	0.9108	**−12236**
KdV	17.34	−16614	9.2987	**−22419**
KS	0.4508	−43121	0.3813	**−43533**
NS	False Eq.	−4379570	11.43	**−28710947**
RD: u_t	3.1177	−14790	2.1639	**−14963**
RD: v_t	3.3251	−15729	2.2967	**−15916**
GS: u_t	0.02621	−106720	0.05542	**−113852**
GS: v_t	0.01108	−114432	0.01096	**−120588**

[1] In this case, we testify that the PDEs discovered by the STRidge algorithm are indecisive: $\partial_t u = -0.9482u\partial_x u$ and $\partial_t u = 0.09918\partial^2_x u - 1.0089u\partial_x u$ when the l_0-penalty hyperparameter (see PDE-FIND [7]) is set to 10^{-1} and 10^{-3}, respectively. Similar results were found in the nPIML paper [12].

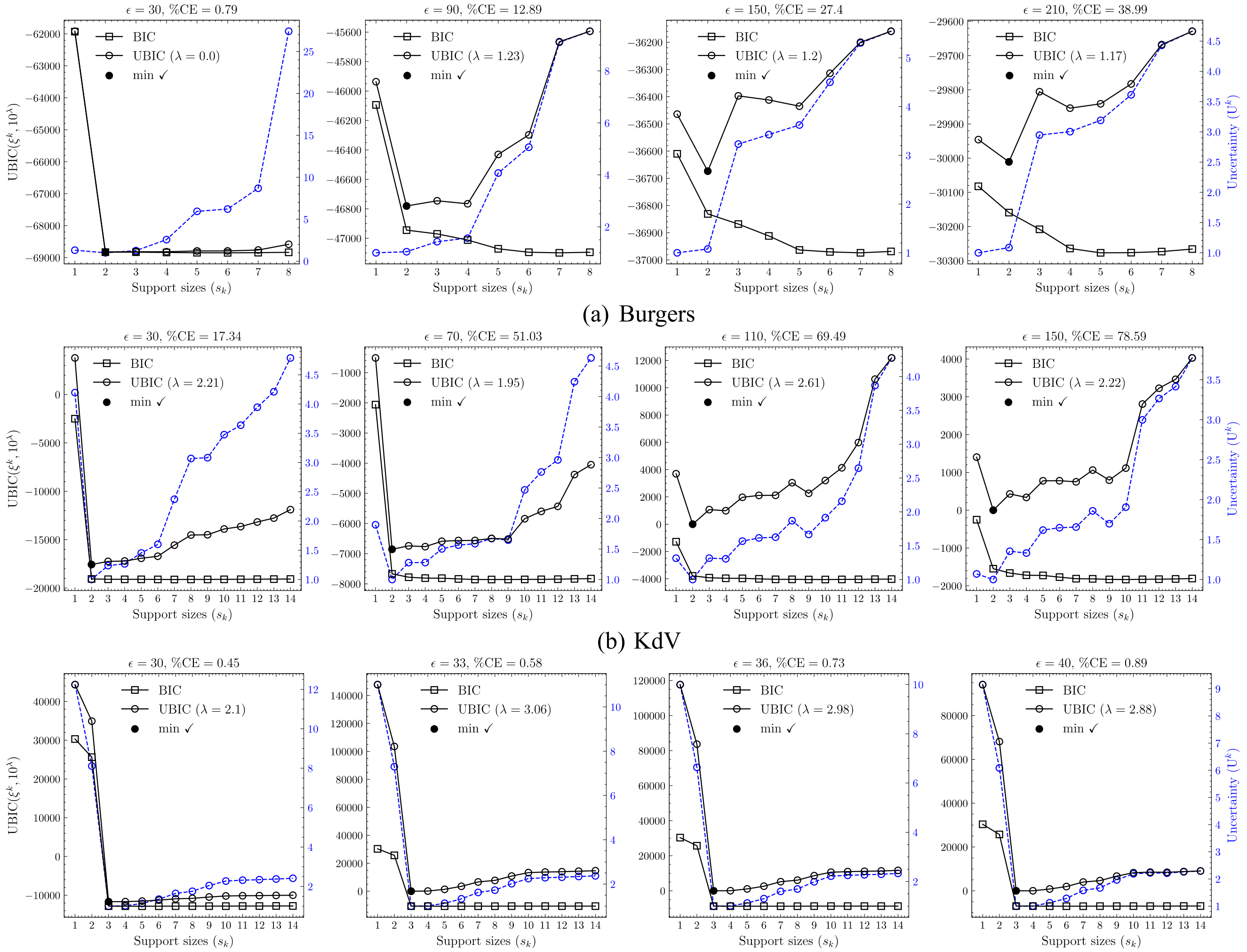
To obtain each noise-reduced variable, we applied 2D Savitzky-Golay filters for spatial denoising at every time step, then employed the denoising SVD. As experimented in the PDE-FIND, we specifically retained the top singular values, which were 26, 20, and 20 for w, u, and v (reshaped as metrics with 325 × 170 rows and 151 columns), respectively. The process enhanced the variables' quality, thereby preserving the correct equation form to be captured at the discovered 4-support-size PDE. The 19 non-weak terms included the following variables: $\psi_1 \in \{w, u, v\}$, the spatial derivatives of the vorticity w: $\psi_2 \in \{w_x, w_{xx}, w_y, w_{yy}\}$, and $\psi_1\psi_2$ (all possible polynomial interactions included). Every best subset, whose cardinality ranges from 1 to 10 was initially approximated by the MIQP (mixed-integer quadratic programming) with the l_0-norm based budget constraint [10]. The computational runtime taken was 99.48 secs. We then performed an all-subsets exhaustive search over the top candidates, each at least existing in one of the 10 best subsets.

Despite the underlying PDE form being dependent on the 4 candidate terms, it is the most stable one, as illustrated in Fig. 3(d) (bottom). Undoubtedly, the quantified uncertainty



(a) Burgers (b) KdV (c) KS

FIGURE 4. Robust adaptive model selection by the UBIC with the preceding denoising step under the extremely noisy scenarios.



(a) Burgers

(b) KdV

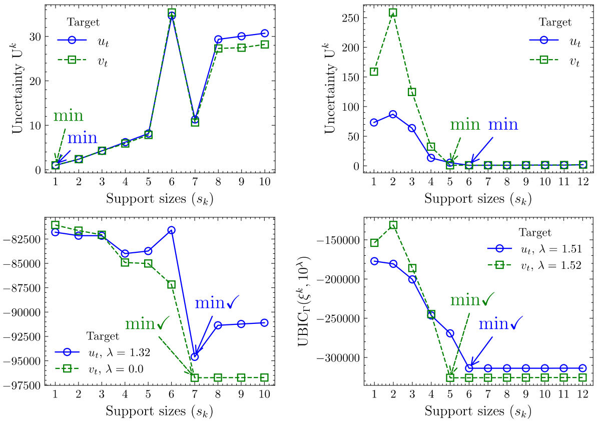
(c) KS

FIGURE 5. Robust model selection by the UBIC without the preceding denoising step under the highly noisy scenarios. For the Burgers and KS cases, we assign $\tau_0 = P_{75}(S)$. For the KdV cases, we assign $\tau_0 = P_{85}(S)$.

associated with each discovered PDE is a beneficial indicator for finding the correct model by the UBIC with $\lambda_U = 10^{6.11}$.

E. DENOISING EFFECT

The positive effect of denoising is evident from the overall drop (shaded area) in BIC, observed throughout the previous examples. We evaluate the trade-off by the maximum reduction in the BIC: $R_{\mathrm{BIC}} = \min_k \mathrm{BIC}(\hat{\xi}^k_\mu) - \max_k \mathrm{BIC}(\hat{\xi}^k_\mu)$ in Table 2. In Fig. 3(c), the reduction in BIC scores is not as pronounced as what is demonstrated in the Burgers and KdV examples, implying the challenge of restoring the chaotic solution of the KS PDE. In the NS example, the omission of



(a) RD

(b) GS

FIGURE 6. Model selection results for the RD and GS PDEs.

the true PDE when no denoising processes were performed causes a noticeable gap in the BIC values between the two cases, as illustrated in Fig. 3(d), confirming the usefulness of the denoising step to the model selection. R_{BIC} or the area between trade-offs is a prospective metric for tuning denoising hyperparameters, possibly facilitating a clearer identification of the governing PDE.

F. ROBUST ADAPTIVE MODEL SELECTION

We fully harness the capability of our proposed method in such extremely noisy scenarios that, without the denoising step, the true governing PDE would be omitted or poorly recovered from the best subsets. Fig. 4 shows that we correctly identify the governing PDEs selected using the adaptive UBIC despite the severe noise interference. Particularly in the Burgers example, neither the BIC nor the uncertainty alone can recover the true equation form.

We also tried ablating the denoising step to solely justify the usability of the UBIC with just the weak formulation and the adaptive model selection. As illustrated in Fig. 5, the correct identification of the governing PDEs emphasizes the advantage of incorporating penalizing uncertainty information for the model selection.

G. DISCOVERY ACCURACY

We evaluate the proximity of each discovered $\hat{\xi}^k_j$ to the ground truth ξ_j by the percentage relative coefficient error: $100 \times \left|\hat{\xi}^k_j - \xi_j\right| / \left|\xi_j\right|$. The %CE reported in Table 2 denotes the average over every effective coefficient. In most of the cases, $\hat{\xi}^k$ obtained on the denoised data delivers a lower %CE than when we omit the denoising step.

H. REACTION-DIFFUSION (RD) PDE

The PDE governs a system that simulates double spiral waves on a periodic domain, consisting of 7 actual terms. To countermeasure 10%-sd noise that perturbed a stack of the u and v variables, 2D Savitzky-Golay filters were employed for spatial denoising at each time step, the results were then collected to construct the noise-reduced data.

The candidate library encompassed the following variables and their transformations: $\psi_1 \in \{u, v, u^3, v^3, u^2v, uv^2\}$, the spatial derivatives (up to second-order) of either u or v: $\psi_2 \in \{u_x, u_y, u_{xx}, u_{yy}, u_{xy}, v_x, v_y, v_{xx}, v_{yy}, v_{xy}\}$, or $\psi_1\psi_2$ (polynomial interaction). To identify the best subset for each cardinality from 1 to 10, an initial approximation was obtained (in $2.56 + 2.62$ secs for u_t and v_t) using the MIQP with the budget constraint based on the l_0-norm. We then ensured the optimality of the subsets with support sizes not greater than 10, respecting the set of unique effective candidates.

As evidenced by Fig. 6(a), the uncertainty positively correlated with the support size, as implied by Equation (8). The uncertainty alone without the base BIC in Equation (6) is thus not enough for model selection. The 1-support-size PDE exhibits the least uncertainty. Nevertheless, an intriguing observation emerges at the 7-support-size PDE, where the uncertainty drops relatively to the surrounding PDEs plotted alongside. This local minimum is exploited by the tuned UBIC with $\lambda_{\mathrm{U}} = 10^{1.32}$ for u_t and $\lambda_{\mathrm{U}} = 10^0$ for v_t to successfully identify the 7 true candidates.

I. GRAY-SCOTT (GS) PDE

The GS PDE governs the reaction-diffusion system in the 4D spatio-temporal grid. For each variable in the noisy stack, we looped through the t-temporal and then z-spatial axes to perform spatial denoising using 2D Savitzky-Golay filters. Hereafter, similarly to the NS example, each variable was reconstructed via the denoising SVD, retaining the 10 most significant singular values.

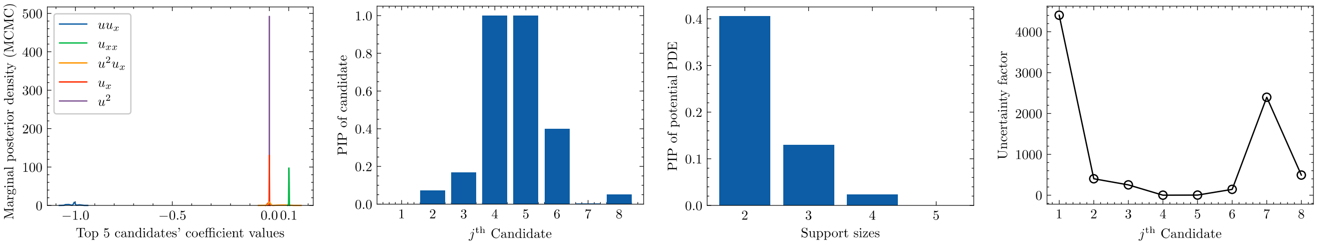
We comprised an overcomplete candidate library with the variables (including a constant term) and their transformations: $\{1, u, v, u^3, v^3, u^2v, uv^2\}$, the spatial derivatives of u: $\{u_x, u_y, u_z, u_{xx}, u_{yy}, u_{zz}, u_{xy}, u_{xz}, u_{yz}\}$, and the spatial derivatives of v: $\{v_x, v_y, v_z, v_{xx}, v_{yy}, v_{zz}, v_{xy}, v_{xz}, v_{yz}\}$. The best subsets were approximated (in $0.18 + 0.18$ secs for u_t and v_t) using the FROLS solver with a maximum support size of 12. These subsets were guaranteed to be at their optimum within all the effective candidates, each once delivered by the solver.

According to Fig. 6(b), it is evident that the best subsets, which align with the true complexity of the PDE system with support sizes of 6 and 5 for u_t and v_t, exhibit the minimal uncertainty values. The tuned UBIC (with $\lambda \approx 1.51$) leverages the uncertainty pattern to penalize the BIC values, hence the successful identification of the true PDE system.

J. COMPARISON WITH CONVENTIONAL BAYESIAN PDE DISCOVERY METHODS

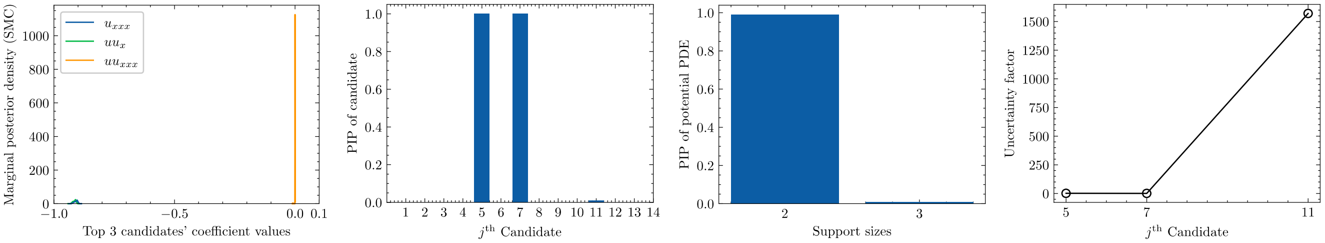
1) UQ-SINDY: SPARSE BAYESIAN REGRESSION BY SPARSIFYING PRIORS

We study the PDE discovery approach based on sparsifying priors, as inspired by [35]. Particularly, we examine sparse



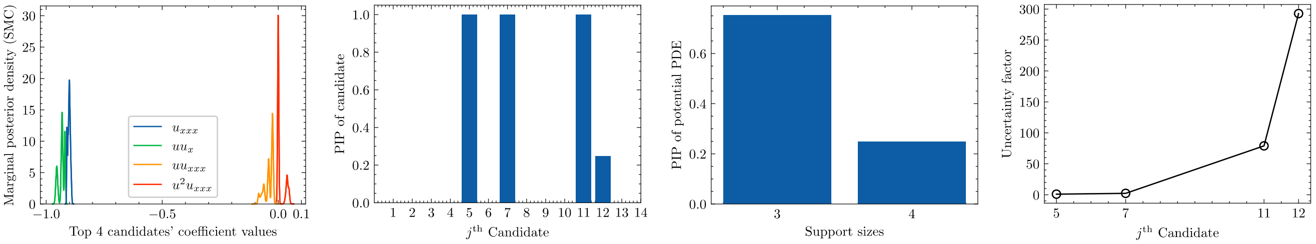
(a) Marginal posterior (b) Candidates' PIP (c) Best subsets' PIP (c) Uncertainty factor

FIGURE 7. Sparse Bayesian regression with the SS prior ($\beta = 0.3125$) for discovering the Burgers' PDE. The candidates are listed in the following order: $[u, u^2, u_x, u_{xx}, uu_x, u^2u_x, uu_{xx}, u^2u_{xx}]$. (a) Marginal posterior density of the top candidates. (b) PIP of the candidates (c) PIP of the best subsets of different support sizes (d) Quantified uncertainty factor of the candidates that have PIPs greater than 0. Note that these comments on (a), (b), (c), and (d) apply to Fig. 8 and Fig. 9.



(a) Marginal posterior (b) Candidates' PIP (c) Best subsets' PIP (c) Uncertainty factor

FIGURE 8. Sparse Bayesian regression with the SS prior ($\beta = 0.01$) for discovering the KdV PDE. The candidates are listed in the following order: $[u, u^2, u_x, u_{xx}, u_{xxx}, u_{xxxx}, uu_x, u^2u_x, uu_{xx}, u^2u_{xx}, uu_{xxx}, u^2u_{xxx}, uu_{xxxx}, u^2u_{xxxx}]$.



(a) Marginal posterior (b) Candidates' PIP (c) Best subsets' PIP (c) Uncertainty factor

FIGURE 9. Sparse Bayesian regression with the SS prior ($\beta = 0.1$) for discovering the KdV PDE. The candidate list is given in Fig. 8.

Bayesian regression with the spike and slap prior (SS) [52], [53], [54] and the regularized horseshoe prior (RH) [55] to solve Equation (4) through Bayesian inference, where we draw samples from the posterior distribution using either Markov chain Monte Carlo (MCMC) or sequential Monte Carlo (SMC) methods implemented in the PyMC package [56].

For the SS prior, each coefficient is given hierarchically as

$$\xi_j^{\text{SS}} \mid \mathcal{B}_j \sim \mathcal{N}(\hat{\xi}_j^b, 1)\mathcal{B}_j;\ \mathcal{B}_j \sim Bernoulli(\beta), \tag{12}$$

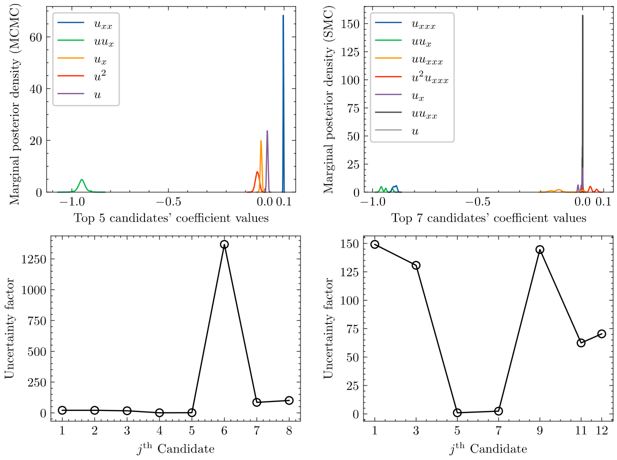
where β is the probability of success of the Bernoulli distribution. As a result that $\mathcal{B}_j \in \{0, 1\}$, the spike and slap prior enables sparse coefficients. Here, b is the index of the user-specified maximum number of effective candidates. Employing the UQ-SINDy based approach, we limit the total number of effective candidates to 8 and 7 for discovering the Burgers' PDE and the KdV PDE, respectively.

After completing the Bayesian inference, the quantified uncertainty factor of each coefficient is given by the following calculation over its posterior samples of $\hat{\xi}_j^{\text{SS}}$:

$$\mathrm{U}_j^{\text{SS}} = \frac{\mathrm{CV}_j^{\text{SS}}}{\min_j \mathrm{CV}_j^{\text{SS}}};\ \mathrm{CV}_j^{\text{SS}} = \frac{\sqrt{\mathbb{V}[\hat{\xi}_j^{\text{SS}}]}}{\mathbb{E}[\hat{\xi}_j^{\text{SS}}]}. \tag{13}$$

Under our core assumption, U_j^{SS} would be comparatively low for the j^{th} effective candidate that corresponds to one of the true terms. For a candidate to be considered effective, its posterior inclusion probability (PIP) must be greater than zero. By counting the number of times a unique subset occurs in the posterior samples, we can also estimate the posterior inclusion probability (PIP) for each (best) subset. For the RH prior, the uncertainty factor U_j^{RH} is computed likewise.

In Fig. 7 and Fig. 8, the desirable outcomes are expected by inspecting that the best subsets with the correct support size



(a) Burgers

(b) KdV

FIGURE 10. **Sparse Bayesian regression with the RH prior for discovering the Burgers and KdV PDEs. The global shrinkage parameter is set to 10^{-3}. The candidate lists are given in Fig. 7 and Fig. 8. In the first row, the marginal posterior density of the top candidates is plotted, while the second row shows the quantified uncertainty factor of the potential posterior candidates.**

(2 in these cases) present the highest PIPs. The marginal posterior of the negligible candidates primarily distributes a spike around the origin, whereas wider or slapped distributions are observed for the true nonzero candidates. However, achieving the results necessitates appropriate settings of the Bernoulli distribution's probability of success in Equation (12): $\beta = 0.3125$ and 0.01 for the Burgers and KdV examples, respectively. If a bigger $\beta = 0.1$ is asserted for the KdV example, the best subset with the highest PIP is comprised of 3 candidates instead, which does not convey the true sparsity as shown in Fig. 9, hence the troublesome sensitivity caused by β. On the contrary, the uncertain pattern remains insensitive to the change in β from 0.01 to 0.1. Also, the u_{xxx} and uu_x (5th and 7th) candidates yield the least uncertainty factor values. The uncertainty factors of u_{xx} and uu_x (4th and 5th candidates) are found to be minimal for the Burgers' PDE case, as illustrated in Fig. 7.

By the RH prior design [35], the coefficients sampled from the posterior distribution are not strictly sparse, exhibiting values close to, but not precisely, zero(s). Thus, a definition of pseudo-probabilities may be introduced. In this study, we are inclined to consider the uncertainty factor as our preferred alternative. Fig. 10 reveals that the true nonzero candidates have the lowest uncertainty factor, akin to the cases where the spike and slap priors are utilized. We set the global shrinkage parameter of the RH prior equal to 10^{-3} for both examples.

2) THRESHOLD SPARSE BAYESIAN REGRESSION

Another approach for achieving the sparse identification of the governing PDE with quantified uncertainty (error bars) is through iterative thresholding until no further changes in sparsity are detected. This approach is known as threshold sparse Bayesian regression [36]. We adhere closely to their iterative thresholding algorithm. In our implementation, we leveraged fast automatic relevance determination (ARD) that uses sparse Bayesian learning [57], [58] to estimate mean regression coefficients from the posterior distribution.

TABLE 3. **Nonoverlapping train and validation domains bounded for performing the PINN-based model selection.**

Dataset	$\mathcal{D}_{\text{Train}}$		$\mathcal{D}_{\text{Val}}$	
	x	t	x	t
Burgers	$[-8, 0]$	$[0, 5]$	$[0.0625, 7.9375]$	$[5.1, 10]$
KdV	$[-20, 19.84]^1$	$[20.08, 40]$	$[-19.92, 19.92]^2$	$[0, 20]$
KS	$[0.0982, 50.36]$	$[0, 50]$	$[50.46, 100.53]$	$[50.4, 100]$

[1] From even indices of the dataset's discretized x.
[2] From odd indices of the dataset's discretized x.

We showcase the experimental results with 3 different pre-specified threshold values, as depicted in Fig. 11. We find the sensitivity with respect to the threshold. Overly high or low threshold values respectively yield underfitted and overfitted PDE models, hence the indecisive model selection results. We address this issue by aggregating the best subsets from the three cases (Threshold $= 10^{-1}, 10^{-2}, 10^{-3}$) and selecting the one that minimizes the UBIC, as seen in Fig. 11(d).

K. SENSITIVITY ANALYSIS

We assess the sensitivity of Algorithm 1 by its success rate in identifying or converging to the true equation under different values of τ_0 and a fixed $N_\delta = 3$. Recall that we mention the two distinct strategies for assigning τ_0 values that result in the selection of PDEs whose support sizes are greater than one: (i) using raw numerical values and (ii) adopting percentiles of successive improvement factors S, considering only BIC-decreasing models. We focus on the following strategies.

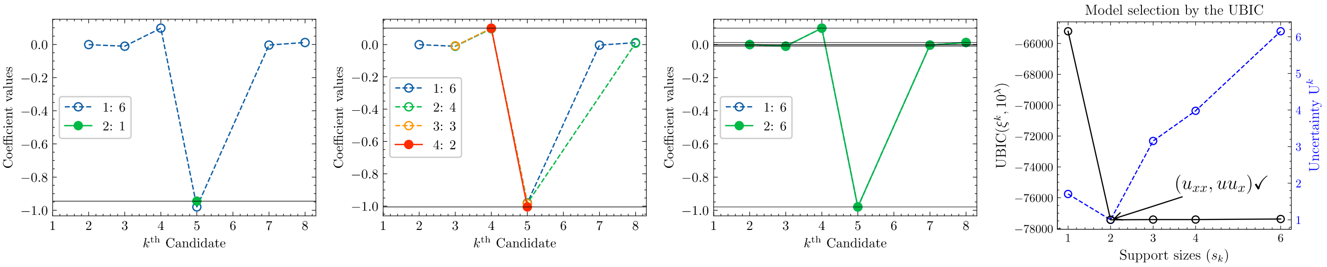
- (i): $\tau_0 = 10^{-3}(1 + h)$; where $h \in \{0, 1, 2, \ldots 99\}$.
- (ii): $\tau_0 = P_i(S)$; where $i \in \{55, 60, 65, \ldots 100\}$.
  P_i calculates the i^{th} percentile of the set S.

The success rate of each strategy is given by the number of times we successfully identify the true equation form over the number of all τ_0 values that lead to the selection of PDEs whose support sizes (s_k) is more than one (to avoid overly high values of τ_0).

We achieve the perfect 100% success rate for some of the examples listed in Table 1. For the examples, in which the success rates are less than 100% using one of the strategies, we create Fig. 12, showing the suggested support sizes (by Algorithm 1) against the τ_0 values within the specified range. In spite of the possible inappropriate uses of excessively low τ_0 that causes the selection of overfitted PDEs, the success rates exceeding 80% are deemed acceptable, implying the low sensitivity of Algorithm 1.

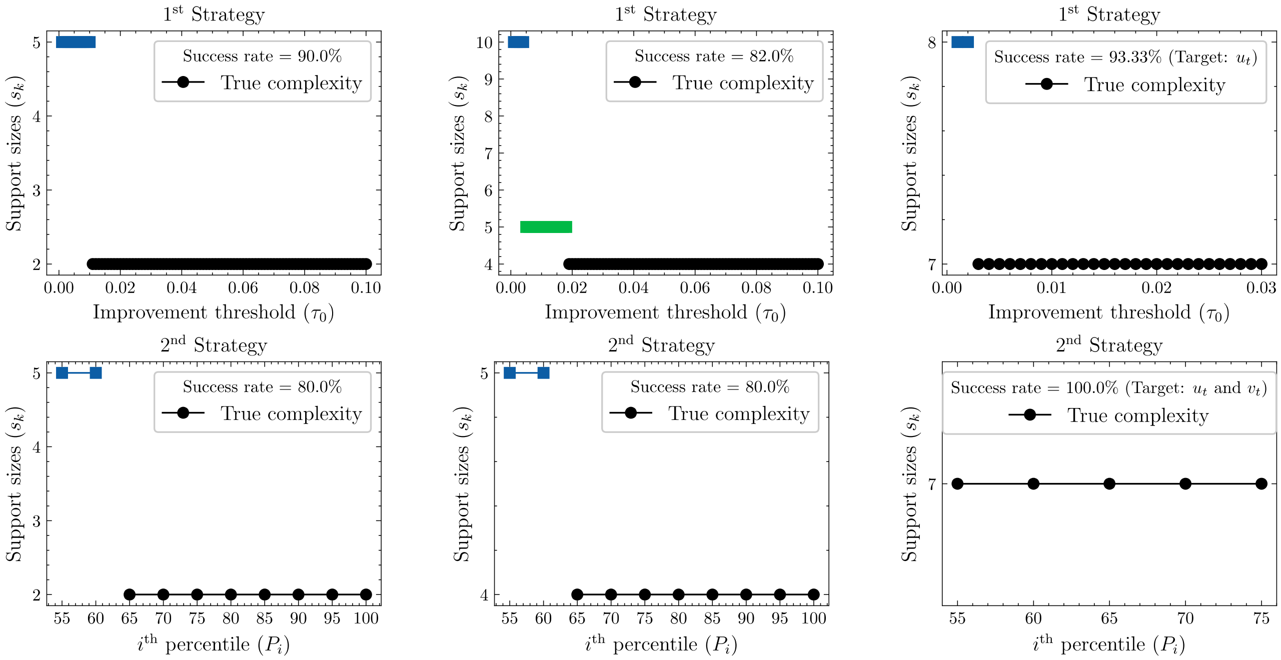
L. SIMULATION-BASED MODEL COMPARISON

We simulated the PDE selected by the tuned UBIC and another potential PDE with an additional candidate, using the PINN learning. The PINN architecture comprised 4 hidden



(a) Threshold = 10^{-1} (b) Threshold = 10^{-2} (c) Threshold = 10^{-3} (d) Uncertainty U^k and UBIC ($\lambda = 0.0$)

FIGURE 11. **Threshold sparse Bayesian regression for discovering Burgers' PDE. The candidate list is given in Fig. 7. The algorithm's threshold is varied to the three different values in (a), (b), and (c). The UBIC scores of the best subsets from the three cases are plotted in (d).**



(a) KdV (b) NS (c) RD

FIGURE 12. **Sensitivity analysis on the KdV, NS, and RD examples, where one (or both) of the success rates of the strategies falls below 100%.**

TABLE 4. **Simulation-based model comparison between the PDEs with (optimal) s_{k^*} and (suboptimal) s_{k^*+1} support sizes.**

Dataset		PINN[1]	Dedalus	Chebfun
Burgers	$s_{k^*} = 2$	$\mathbf{-16439}$	$\mathbf{-134490}$	$\mathbf{-134490}$
	$s_{k^*+1} = 3$	−2020	−134150	−134160
KdV	$s_{k^*} = 2$	$\mathbf{247070}$	$\mathbf{-732057}^2$	$\mathbf{-708493}^2$
	$s_{k^*+1} = 3$	280799	-729386^2	Divergence[3]
KS	$s_{k^*} = 3$	$\mathbf{339036}$	$\mathbf{783459}$	$\mathbf{783461}$
	$s_{k^*+1} = 4$	501440	811910	811906

[1]The simulated solution by PINN is evaluated on the validation set $\mathcal{D}_{\text{Val}}$ detailed in Table 3. [2]Before the simulation, the PDE coefficients are refitted using CWS [34] then added with a small bias value of -10^{-4}, which minimizes the resultant BIC scores. [3]The solution obtained by the spin function explodes (diverges) with a time-step of 10^{-5}.

layers, each with 5 neurons. The learning rate parameter of the L-BFGS optimization algorithm was initialized equal to 0.1. The number of training epochs was set to 500 (taking approximately less than 1.5 hours when training on a Quadro RTX graphics processing unit with 49152 MiB memory to converge to the local optimum).

Comparing the two PDEs, we measured the proximity of their simulated solutions to the denoised observed data using the BIC in Equation (11). Table 4 warrants that the s_{k^*}-support-size PDE has indeed the sufficient complexity in yielding the lower-BIC simulated state variable than its competitor, the s_{k^*+1}-support-size PDE with the dispensable candidate. Ensuring our findings, we also solve the PDEs on their entire spatio-temporal domain using the Chebfun and Dedalus software, which is unlike the PINN approach that necessitates a train-validation data split. The symbolic representation of the initial condition required by the software is recovered using the PySR package [59]. In addition, symbolic regression with searching for simplifying properties (see AI Feynman [1]) can be useful for understanding the boundary condition, which may further help us decide on the governing PDE.

TABLE 5. Frobenius and infinity matrix norm-based errors (l_F and l_∞) between the simulated solution of the UBIC-selected PDE and the noiseless PDE solution.

Dataset		PINN[1]	Dedalus	Chebfun	Denoised $\hat{u}$
Burgers	l_F	**0.335944**	0.481315	0.481312	2.85836
	l_∞	$\underline{0.293584}$	0.485626	0.485622	2.27481
KdV	l_F	29.1799[2]	**7.21427**	11.7038	28.3743
	l_∞	36.4070[2]	$\underline{8.03854}$	14.5979	41.0023
KS	l_F	27.4813	562.835	562.838	**27.1081**
	l_∞	29.4790	327.662	327.490	$\underline{17.8813}$

[1]The PINN learning without optimizing $\hat{\xi}^{\kappa}$ ($\kappa = k^*$) is conducted on the entire domain. Only for the KS case, the number of neurons per layer is 50, and pretraining the network to fit the denoised observed data without minimizing any physical constraint is undergone. [2]Before the PINN-based simulation, the PDE coefficients are just refitted using CWS [34].

In Table 5, we evaluate the UBIC-selected PDE by the Frobenius and infinity matrix norms (l_F and l_∞) of the difference between its simulated solution to the noiseless PDE solution. For the Burgers and KdV PDEs, the results confirm that the simulated solution of the UBIC-selected PDE captures the noiseless solution. For the KS PDE, it is more difficult to simulate the chaotic solution better than the denoised $\hat{u}$ even though the slightly inaccurate PDE coefficients (with %CE $=$ 0.38) are used. Regarding the simulation methods, the PINN-based solver, learning from both the physical constraint and the denoised observed data $\hat{u}$, performs competitively to the Dedalus or Chebfun software. Our best-simulated solutions are visualized in Fig. 14 (see Appendix A).

V. CONCLUSION

A. SUMMARY

We extend the BIC to the parameter-adaptive UBIC, which incorporates the model uncertainty for selecting the governing PDE amidst noisy data. The quantified uncertainty from the model coefficient posterior promotes the reliability of the model selection, preventing the selection of overfitted PDEs unaddressed by the BIC. Thanks to the derived analytical posterior, the proposed UBIC quickly demonstrates the successful identification of the true hidden equation, taking less than 0.1 secs except for the NS case, where Algorithm 1 completes in 17.2 secs. Validating consistency in model selection results, we perform a comparison between the PDE selected by the UBIC and its competitor with an extra candidate to choose the better PDE that delivers a lower BIC value calculated between the PINN-simulated state and the denoised observed data. Finally, we show that the PDE discovery from denoised data positively improves the BIC trade-off.

B. LIMITATIONS AND FUTURE WORK

For future work, we encourage the development of data-driven PDE discovery in the following directions, which are geared toward addressing existing limitations.

- Relaxing the overcomplete assumption by gradually improving a set of candidates instead of keeping a large candidate library unvaried during the optimization process. The concept involves removing bad candidate functions and introducing new ones to refine the set of candidates.
- Robust PDE discovery method against different types of noise to compensate for not having access to good-quality training data.
- Accelerated computation by reducing a candidate library size (search space) before running sparse or best-subset regression solvers.

For example, researchers may consider employing the UBIC with genetics algorithms [60] to discover parametric PDEs with temporal or spatial dependence under the relaxed overcompleteness assumption.

APPENDIX A
DISCOVERY OF VISCOUS BURGERS' PDE WITH SHOCK WAVES

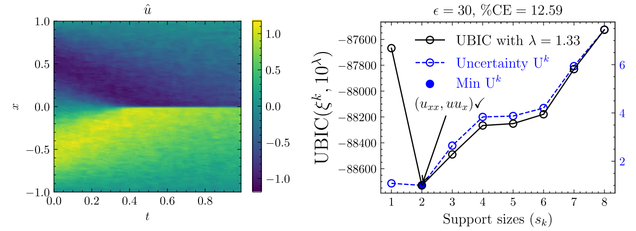
Developing upon the Burgers' PDE studied in the main text, we investigated our proposed approach against shock waves by considering a smaller fluid viscosity of $\frac{0.01}{\pi} \approx 0.003183$ instead of 0.1. The clean PDE dataset utilized in this section was borrowed from [21]. The system state variable perturbed by 30%-sd noise was regarded as the noisy observed data.

Under the same experimental setting, the best subsets are retrieved in 0.30 secs, and then our PDE discovery approach successfully identifies the governing Burgers' PDE form with a good %CE of 12.59 despite the abrupt transition (discontinuity), as depicted Fig. 13. The UBIC with $\lambda_{\text{U}} = 10^{1.33}$ correctly selects the PDE read as follows: $\partial_t u = 0.003746\partial_x^2 u - 0.925141u\partial_x u$.

The PINN-simulated solution of the discovered 2-support-size PDE (BIC $=$ 15101) is better than that of the discovered 3-support-size PDE (BIC $=$ 44666) on the validation set bounded by $x \in [\frac{3}{255}, 1]$ and $t \in [0.51, 0.99]$. The Dedalus software endorses our decision on selecting the best PDE, providing the BIC scores of -67174 and -36774 for the PDEs with 2 and 3 support sizes, respectively. As shown in Fig. 14(d), the simulated solution of the UBIC-selected PDE resembles the clean solution with $l_F = 4.96916, 3.08782, 3.09236$ and $l_\infty = 7.81370, 5.80897, 5.94630$ using the PINN learning, the Dedalus software, and the Chebfun software, respectively.

APPENDIX B
UNCERTAINTY-PENALIZED WAIC (UWAIC)

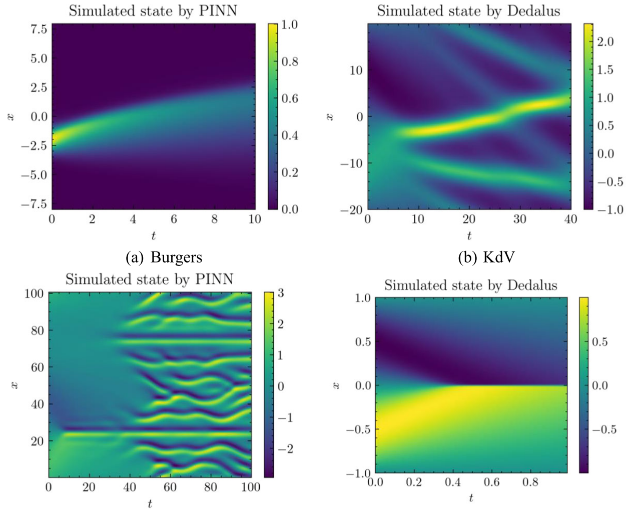
We conduct a pilot extension of the uncertainty penalization to the WAIC (widely applicable information criterion) [61], defining $\text{UWAIC} = \text{WAIC} + \lambda_{\text{U}}\text{CV}^k$; where $\text{WAIC} = \text{BL} + \frac{\text{FV}}{N_\Omega}$. BL and FV are the Bayes training loss and the functional variance, respectively. UWAIC is more computationally expensive than the proposed UBIC because it involves full Bayesian inference to compute the WAIC before adding the penalizing unnormalized uncertainty CV^k. The UWAIC with its λ_{U} specified in Figure 15 is capable of identifying the



(a) Denoised observed data

(b) Viscous Burgers' PDE with shock waves

FIGURE 13. Model selection results for the Burgers' PDE with shock waves.



(a) Burgers

(b) KdV

(c) KS

(d) Burgers' PDE with shock waves

FIGURE 14. Our best-simulated solution of each UBIC-selected PDE.

true PDE terms, similar to the UBIC. These results suggest the possibility of incorporating the proposed uncertainty penalization into other related information criteria.

APPENDIX C
DENOISING METHOD COMPARISON

To assure the superiority of the K-SVD algorithm over traditional denoising methods, such as those based on SVD, PCA, and wavelet transformation, we conducted a comparative analysis of the quality of denoised observed data in Table 6. The results reveal that the wavelet transform-based method can generate denoised data that resembles the noiseless PDE solution, but the %CE of the UBIC-selected PDE derived from the denoised data is inaccurate. On the other hand, denoised data obtained using SVD or PCA based methods may result in an accurate UBIC-selected PDE, but the error from the noiseless solution is high, possibly leading to inaccurate symbolic recovery of the initial and boundary conditions. Considering each evaluation metric, the K-SVD algorithm demonstrates the best performance on average (across the datasets), outperforming both the robust PCA and the denoising discrete Fourier transform (DFT). Therefore, the K-SVD algorithm is our preferred choice for denoising 2D spatio-temporal (image-like) data.

TABLE 6. Each denoising method is evaluated by two metrics: (i) the Frobenius matrix norm error (l_F) between the noiseless PDE solution and the denoised observed data produced by the method, and (ii) the %CE of the UBIC-selected PDE based on the denoised data.

Method		Burgers	KdV	KS
Regularized K-SVD	l_F	2.85836	28.3743	**27.1081**
	%CE	0.910797	9.29866	0.381328
SVD 90% explained var.[1]	l_F	4.76071	51.4422	80.9308
	%CE	Failed[2]	Failed	11.8720
SVD 95% explained var.	l_F	3.96137	67.7331	103.447
	%CE	2.38539	13.6785	0.407940
SVD 99.99% explained var.	l_F	8.66730	107.199	161.975
	%CE	0.778044	17.2959	0.446532
PCA 90% explained var.	l_F	4.75426	51.6717	81.0748
	%CE	Failed	Failed	10.4475
PCA 95% explained var.	l_F	5.53985	67.3033	102.736
	%CE	1.48056	13.7889	0.176340
PCA 99.5% explained var.	l_F	8.41187	103.955	157.161
	%CE	0.521065	16.5462	0.481844
Robust PCA [14]	l_F	7.44519	104.832	161.867
	%CE	1.95383	16.8732	0.452507
Discrete Fourier Transform[3]	l_F	7.99949	98.7912	148.519
	%CE	0.764020	17.4676	0.339274
Wavelet transform[4] [62]	l_F	**2.52871**	**24.2224**	62.4467
	%CE	2.49423	14.4260	0.862688

[1] We keep the minimum number of components that can retain the explained variance. [2] With $\tau_0 = P_{75}(S)$, incorrect candidates are selected. [3] Frequency components with a PSD (power spectral density) below the 50th percentile are filtered out. [4] Image denoising using Bayes shrink thresholding based wavelet transformation.

APPENDIX D
COMPARISON WITH CONVENTIONAL MODEL SELECTION METHODS

A. CROSS-VALIDATION BASED MODEL SELECTION

One might presume that the cross-validation strategy would suffice for selecting the optimal support size from Eq. (4). To examine this assumption, we conducted experiments using various data splitters available in the Scikit-learn package [63]. We experimented with the following splitter classes.

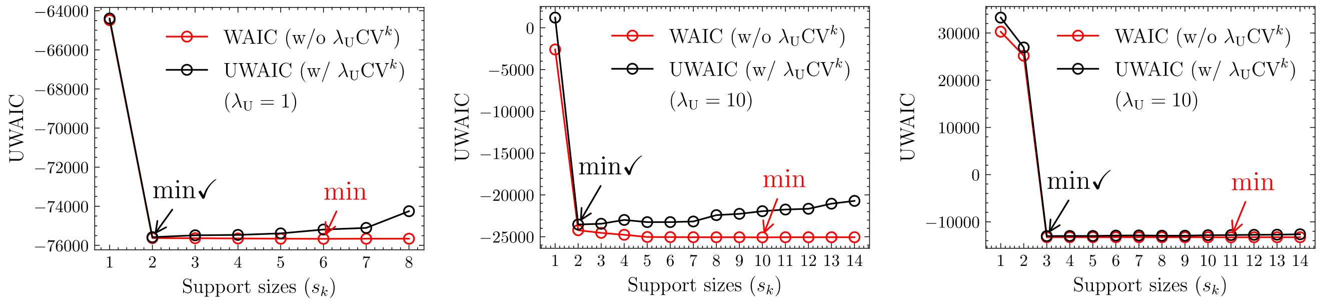
- `RepeatedKFold(n_splits=2, n_repeats=5)`
- `ShuffleSplit(n_splits=10, test_size=0.5)`

The former splitter was configured with 2-folds and repeated 5 times with different randomization in each repetition. Contrarily, the latter splitter was configured with 10 re-shuffling and splitting iterations but did not guarantee that all folds were different. We also employed the single train-validation split procedure, wherein the first half of the samples were allocated for training and the second half for validation.

In the cases of Burgers and KS, as shown in Fig. 16, the use of validation sets safeguards against the wrong selection of the overfitted PDEs. Nevertheless, none of the splitters was found effective for the KdV example, where there are noticeable BIC drops observed during the transition from the 2-support-size PDE to the 5-support-size PDE. We call this scenario the "cross-validation pitfalls," which could have misinformed us that the support size of 5 is optimal.

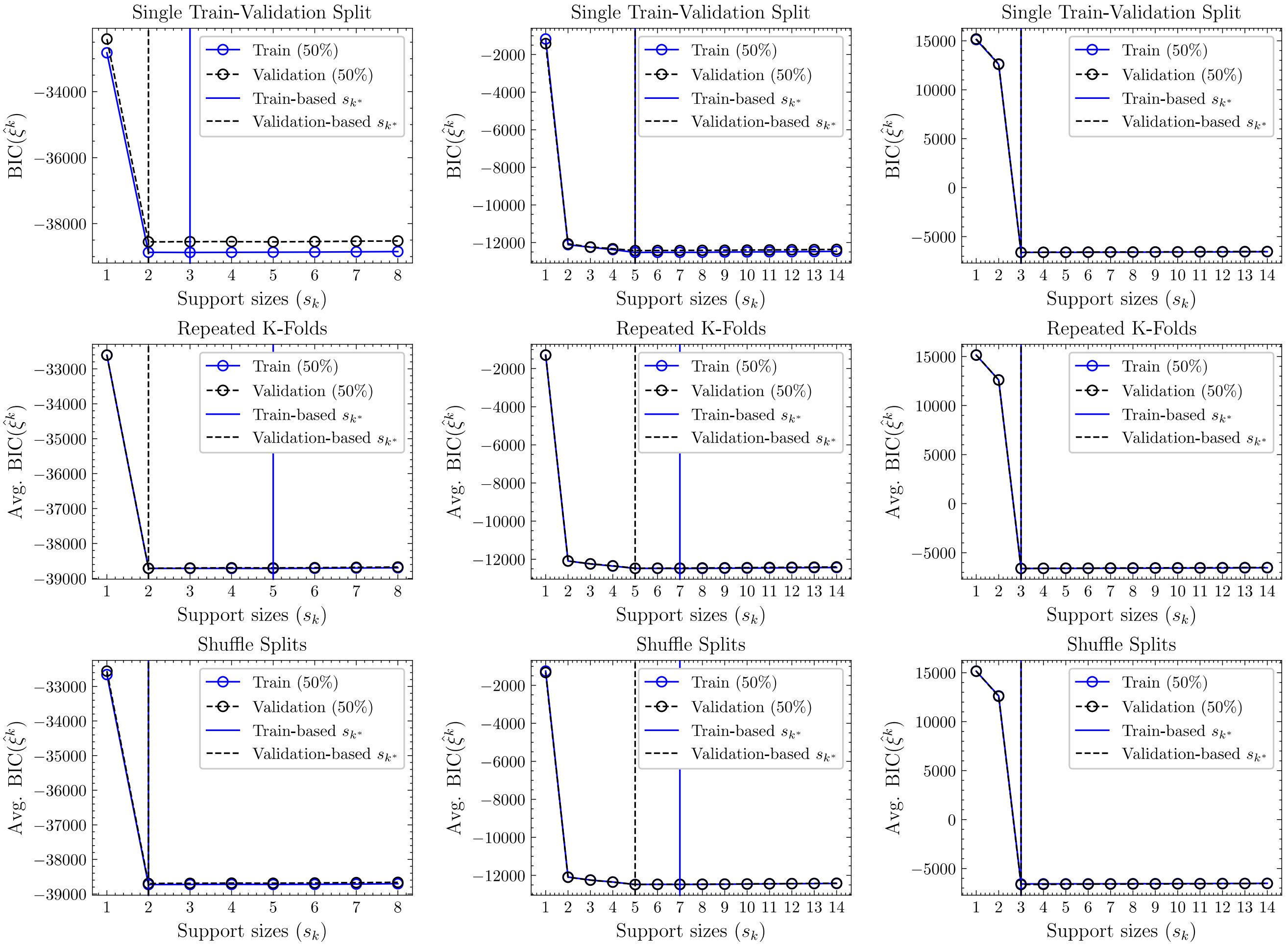
B. CANDIDATE IMPORTANCE FOR MODEL SELECTION

We explored whether importance scores of candidate terms could guide the discovery of the true terms constituting



(a) Burgers (b) KdV (c) KS

FIGURE 15. Model selection results by the uncertainty-penalized WAIC (UWAIC).

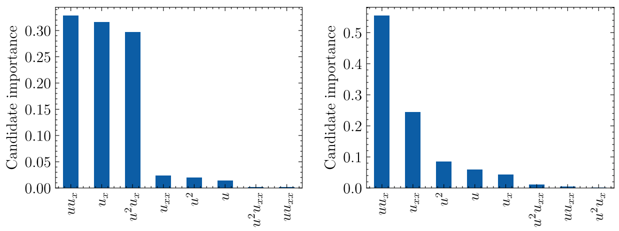


(a) Burgers (b) KdV (c) KS

FIGURE 16. Using cross-validation based model selection to find the optimal support sizes for Burgers, KdV, and KS examples. "Avg." is an abbreviation for "average."

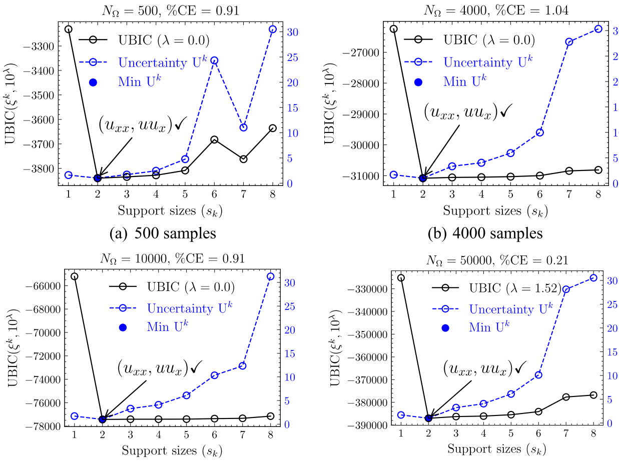
the governing PDE. We studied two distinct assignments of the importance of a candidate: (i) the coefficient of determination (R-squared) of a single-regressor (univariable) linear model and (ii) the mean absolute SHAP (Shapley additive explanations) value [64] of a linear model fitted on every candidate.

In Fig. 17, we rank the candidates in the Burgers' PDE example. The R-squared concerning one regressor ranks the variables uu_x and u_x as the top two single predictors. However, the top single predictors do not necessarily compose the governing PDE. It is noteworthy that using this metric for candidate elimination may risk discarding a



(a) R-squared (b) SHAP

FIGURE 17. Ranking the candidates in Burgers' PDE (without no shock wave) by the R-squared and the mean absolute SHAP value (stated in Appendix D-B). The candidate importance scores are normalized such that the total summation is 1.



(a) 500 samples (b) 4000 samples

(c) 10000 samples (d) 50000 samples

FIGURE 18. For discovering the Burgers' PDE (in the main text), the model selection results are provided for the different sample sizes.

true candidate (e.g., u_{xx}) that is not predictive when used alone. The SHAP values of the multivariable model yield a satisfactory result, ranking the variables uu_x and u_{xx} as the top two candidates. Unfortunately, neither of the candidate rankings by either assignment is enough to entail the optimal complexity of the governing equation.

C. F-TEST FOR COMPARING TWO MODELS

We explored whether the F-test (implemented in the statsmodels's `anova_lm` function [65]) could be a beneficial tool to determine the optimal complexity of the governing PDE in the Burgers' PDE example in the main text. Note that 2-support-size and 3-support-size PDEs we discovered were composed of variables from $\{u_{xx}, uu_x\}$ and $\{u_x, u_{xx}, uu_x\}$, respectively. Thus, we used the F-test to statistically test if adding the u_x candidate to the best subset of 2 supports results in significant improvement in estimating the weak form q_0. The resulting F-statistic and p-value are 11.3496 and 0.000757 (< 0.001) in order, persuading that we reject the null hypothesis stating that the 3-support-size PDE does not perform significantly better than 2-support-size PDE. Such results could have misled us to select an overfitted PDE. Therefore, we cannot recommend relying on the F-test result to determine when to stop increasing the model complexity.

APPENDIX E
SAMPLE SIZE EFFECTS ON MODEL SELECTION

In this section, we utilize the Burgers example in the main text to show that the proposed UBIC can select the true governing PDE despite the varying sample size N_Ω (representing the number of domain centers when applying the weak formulation in this example), ranging from 500, 4000, 10000, to 50000. A consistent pattern in uncertainty values favoring the correct support size of 2 is observed. Remarkably, with the sample size up to 50000, the %CE of the estimated coefficients is impressively reduced to 0.21.

REFERENCES

[1] S.-M. Udrescu and M. Tegmark, "AI Feynman: A physics-inspired method for symbolic regression," *Sci. Adv.*, vol. 6, no. 16, Apr. 2020, Art. no. eaay2631.

[2] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proc. Nat. Acad. Sci. USA*, vol. 113, no. 15, pp. 3932–3937, Apr. 2016.

[3] S. Li, E. Kaiser, S. Laima, H. Li, S. L. Brunton, and J. N. Kutz, "Discovering time-varying aerodynamics of a prototype bridge by sparse identification of nonlinear dynamical systems," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 100, no. 2, Aug. 2019, Art. no. 022220.

[4] J. H. Lagergren, J. T. Nardini, G. Michael Lavigne, E. M. Rutter, and K. B. Flores, "Learning partial differential equations for biological transport models from noisy spatio-temporal data," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 476, no. 2234, Feb. 2020, Art. no. 20190800.

[5] S. Lee, Y. M. Psarellis, C. I. Siettos, and I. G. Kevrekidis, "Learning black- and gray-box chemotactic PDEs/closures from agent based Monte Carlo simulation data," *J. Math. Biol.*, vol. 87, no. 1, p. 15, Jul. 2023.

[6] J. Horrocks and C. T. Bauch, "Algorithmic discovery of dynamic models from infectious disease data," *Sci. Rep.*, vol. 10, no. 1, p. 7061, Apr. 2020.

[7] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Data-driven discovery of partial differential equations," *Sci. Adv.*, vol. 3, no. 4, Apr. 2017, Art. no. e1602614.

[8] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc., Ser. B, Methodolog.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.

[9] P. Zheng, T. Askham, S. L. Brunton, J. N. Kutz, and A. Y. Aravkin, "A unified framework for sparse relaxed regularized regression: SR3," *IEEE Access*, vol. 7, pp. 1404–1423, 2019.

[10] D. Bertsimas, A. King, and R. Mazumder, "Best subset selection via a modern optimization lens," *Ann. Statist.*, vol. 44, no. 2, pp. 813–852, Apr. 2016.

[11] D. Bertsimas and W. Gurnee, "Learning sparse nonlinear dynamics via mixed-integer optimization," *Nonlinear Dyn.*, vol. 111, no. 7, pp. 6585–6604, Apr. 2023.

[12] P. Thanasutives, T. Morita, M. Numao, and K.-I. Fukui, "Noise-aware physics-informed machine learning for robust PDE discovery," *Mach. Learn., Sci. Technol.*, vol. 4, no. 1, Feb. 2023, Art. no. 015009.

[13] L. Sun, D. Huang, H. Sun, and J.-X. Wang, "Bayesian spline learning for equation discovery of nonlinear dynamics with quantified uncertainty," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 6927–6940.

[14] J. Li, G. Sun, G. Zhao, and H. L. Li-Wei, "Robust low-rank discovery of data-driven partial differential equations," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 1, 2020, pp. 767–774.

[15] B. Dumitrescu and P. Irofti, "Regularized K-SVD," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 309–313, Mar. 2017.

[16] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Autom. Control*, vol. IT-19, no. 6, pp. 716–723, Dec. 1974.

[17] N. M. Mangan, J. N. Kutz, S. L. Brunton, and J. L. Proctor, "Model selection for dynamical systems via sparse regression and information criteria," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 473, no. 2204, Aug. 2017, Art. no. 20170009.

[18] X. Dong, Y.-L. Bai, Y. Lu, and M. Fan, "An improved sparse identification of nonlinear dynamics with Akaike information criterion and group sparsity," *Nonlinear Dyn.*, vol. 111, no. 2, pp. 1485–1510, Jan. 2023.

[19] G. Schwarz, "Estimating the dimension of a model," *Ann. Statist.*, vol. 6, no. 2, pp. 461–464, Mar. 1978.

[20] Y. Chen, Y. Luo, Q. Liu, H. Xu, and D. Zhang, "Symbolic genetic algorithm for discovering open-form partial differential equations (SGA-PDE)," *Phys. Rev. Res.*, vol. 4, no. 2, Jun. 2022, Art. no. 023174.

[21] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, Feb. 2019.

[22] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators," *Nature Mach. Intell.*, vol. 3, no. 3, pp. 218–229, Mar. 2021.

[23] S. Wang, H. Wang, and P. Perdikaris, "Learning the solution operator of parametric partial differential equations with physics-informed DeepONets," *Sci. Adv.*, vol. 7, no. 40, Oct. 2021, Art. no. eabi8605.

[24] Y. Chen, B. Hosseini, H. Owhadi, and A. M. Stuart, "Solving and learning nonlinear PDEs with Gaussian processes," *J. Comput. Phys.*, vol. 447, Dec. 2021, Art. no. 110668.

[25] G. Pang, L. Yang, and G. E. Karniadakis, "Neural-net-induced Gaussian process regression for function approximation and PDE solution," *J. Comput. Phys.*, vol. 384, pp. 270–288, May 2019.

[26] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Machine learning of linear differential equations using Gaussian processes," *J. Comput. Phys.*, vol. 348, pp. 683–693, Nov. 2017.

[27] H. Xu, J. Zeng, and D. Zhang, "Discovery of partial differential equations from highly noisy and sparse data with physics-informed information criterion," *Research*, vol. 6, p. 147, Jan. 2023.

[28] J. K. Lindsey, "Relationships among sample size, model selection and likelihood regions, and scientifically important differences," *J. Roy. Stat. Soc., Ser. D, Statistician*, vol. 48, no. 3, pp. 401–411, Sep. 1999.

[29] S. Lee, M. Kooshkbaghi, K. Spiliotis, C. I. Siettos, and I. G. Kevrekidis, "Coarse-scale PDEs from fine-scale observations via machine learning," *Chaos, Interdiscipl. J. Nonlinear Sci.*, vol. 30, no. 1, Jan. 2020, Art. no. 013141.

[30] E. Galaris, G. Fabiani, I. Gallos, I. Kevrekidis, and C. Siettos, "Numerical bifurcation analysis of PDEs from lattice Boltzmann model simulations: A parsimonious machine learning approach," *J. Sci. Comput.*, vol. 92, no. 2, p. 34, Aug. 2022.

[31] W. Cao and W. Zhang, "Machine learning of partial differential equations from noise data," *Theor. Appl. Mech. Lett.*, vol. 13, no. 6, Nov. 2023, Art. no. 100480.

[32] F. P. Kemeth, T. Bertalan, T. Thiem, F. Dietrich, S. J. Moon, C. R. Laing, and I. G. Kevrekidis, "Learning emergent partial differential equations in a learned emergent space," *Nature Commun.*, vol. 13, no. 1, p. 3318, Jun. 2022.

[33] P. A. K. Reinbold, D. R. Gurevich, and R. O. Grigoriev, "Using noisy or incomplete data to discover models of spatiotemporal dynamics," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 101, no. 1, Jan. 2020, Art. no. 010203.

[34] D. A. Messenger and D. M. Bortz, "Weak SINDy for partial differential equations," *J. Comput. Phys.*, vol. 443, Oct. 2021, Art. no. 110525.

[35] S. M. Hirsh, D. A. Barajas-Solano, and J. N. Kutz, "Sparsifying priors for Bayesian uncertainty quantification in model discovery," *Roy. Soc. Open Sci.*, vol. 9, no. 2, Feb. 2022, Art. no. 211823.

[36] S. Zhang and G. Lin, "Robust data-driven discovery of governing physical laws with error bars," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 474, no. 2217, Sep. 2018, Art. no. 20180305.

[37] K. L. Ratzlaff and J. T. Johnson, "Computation of two-dimensional polynomial least-squares convolution smoothing integers," *Anal. Chem.*, vol. 61, no. 11, pp. 1303–1305, Jun. 1989.

[38] J. Krumm, *Savitzky–Golay Filters for 2D Images*. Redmond, WA, USA: Microsoft Research, Microsoft Corporation, 2001.

[39] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.

[40] B. de Silva, K. Champion, M. Quade, J.-C. Loiseau, J. Kutz, and S. Brunton, "PySINDy: A Python package for the sparse identification of nonlinear dynamical systems from data," *J. Open Source Softw.*, vol. 5, no. 49, p. 2104, May 2020.

[41] A. Kaptanoglu, B. de Silva, U. Fasel, K. Kaheman, A. Goldschmidt, J. Callaham, C. Delahunt, Z. Nicolaou, K. Champion, J.-C. Loiseau, J. Kutz, and S. Brunton, "PySINDy: A comprehensive Python package for robust sparse system identification," *J. Open Source Softw.*, vol. 7, no. 69, p. 3994, Jan. 2022.

[42] D. Bertsimas and R. Weismantel, *Optimization Over Integers*. Waltham, MA, USA: Dynamic Ideas, 2005.

[43] S. Chen, S. A. Billings, and W. Luo, "Orthogonal least squares methods and their application to non-linear system identification," *Int. J. Control*, vol. 50, no. 5, pp. 1873–1896, Nov. 1989.

[44] S. A. Billings, *Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains*. Hoboken, NJ, USA: Wiley, 2013.

[45] H. Hazimeh, R. Mazumder, and A. Saab, "Sparse regression at scale: Branch-and-bound rooted in first-order optimization," *Math. Program.*, vol. 196, nos. 1–2, pp. 347–388, Nov. 2022.

[46] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.

[47] Z. Battles and L. N. Trefethen, "An extension of MATLAB to continuous functions and operators," *SIAM J. Sci. Comput.*, vol. 25, no. 5, pp. 1743–1770, Jan. 2004.

[48] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown, "Dedalus: A flexible framework for numerical simulations with spectral methods," *Phys. Rev. Res.*, vol. 2, no. 2, Apr. 2020, Art. no. 023068.

[49] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Math. Program.*, vol. 45, nos. 1–3, pp. 503–528, Aug. 1989.

[50] P. Thanasutives, M. Numao, and K.-I. Fukui, "Adversarial multi-task learning enhanced physics-informed neural networks for solving partial differential equations," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–9.

[51] C. Bajaj, L. McLennan, T. Andeen, and A. Roy, "Recipes for when physics fails: Recovering robust learning of physics informed neural networks," *Mach. Learn., Sci. Technol.*, vol. 4, no. 1, Feb. 2023, Art. no. 015013.

[52] H. Ishwaran and J. S. Rao, "Spike and slab variable selection: Frequentist and Bayesian strategies," *Ann. Statist.*, vol. 33, no. 2, pp. 730–773, Apr. 2005.

[53] D. Madigan and A. E. Raftery, "Model selection and accounting for model uncertainty in graphical models using Occam's window," *J. Amer. Stat. Assoc.*, vol. 89, no. 428, pp. 1535–1546, Dec. 1994.

[54] T. J. Mitchell and J. J. Beauchamp, "Bayesian variable selection in linear regression," *J. Amer. Stat. Assoc.*, vol. 83, no. 404, pp. 1023–1032, 1988.

[55] J. Piironen and A. Vehtari, "Sparsity information and regularization in the horseshoe and other shrinkage priors," *Electron. J. Statist.*, vol. 11, no. 2, pp. 5018–5051, Jan. 2017.

[56] J. Salvatier, T. V. Wiecki, and C. Fonnesbeck, "Probabilistic programming in Python using PyMC3," *PeerJ Comput. Sci.*, vol. 2, p. e55, Apr. 2016.

[57] M. E Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," in *Proc. Int. Workshop Artif. Intell. Statist.*, 2003, pp. 276–283.

[58] A. Faul and M. Tipping, "Analysis of sparse Bayesian learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 14, T. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA, USA: MIT Press, 2001, pp. 383–389. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2001/file/02b1be0d48924c327124732726097157-Paper.pdf

[59] M. Cranmer, "Interpretable machine learning for science with PySR and SymbolicRegression.jl," 2023, *arXiv:2305.01582*. [Online]. Available: https://arxiv.org/abs/2305.01582

[60] H. Xu, H. Chang, and D. Zhang, "DLGA-PDE: Discovery of PDEs with incomplete candidate library via combination of deep learning and genetic algorithm," *J. Comput. Phys.*, vol. 418, Oct. 2020, Art. no. 109584.

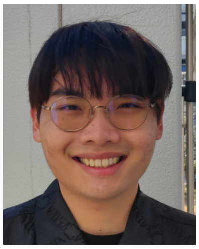
[61] S. Watanabe, "Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory," *J. Mach. Learn. Res.*, vol. 11, no. 116, pp. 3571–3594, 2010.

[62] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "Scikit-image: Image processing in Python," *PeerJ*, vol. 2, p. e453, Jun. 2014.

[63] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12 no. 10, pp. 2825–2830, 2012.

[64] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett, Eds. Red Hook, NY, USA: Curran, 2017, pp. 4765–4774. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[65] S. Seabold and J. Perktold, "Statsmodels: Econometric and statistical modeling with Python," in *Proc. Python Sci. Conf.*, 2010, Art. no. 25080.



PONGPISIT THANASUTIVES received the master's degree in information science and technology from Osaka University, Japan, in 2022, where he is currently pursuing the Ph.D. degree with the Graduate School of Information Science and Technology. His research interests include artificial intelligence, machine learning, neural networks, and the applications to physical sciences.

TAKASHI MORITA received the Ph.D. degree in linguistics from the Massachusetts Institute of Technology in 2018. He then performed postdoctoral research at the Primate Research Institute of Kyoto University from 2018 to 2020, and worked as an Assistant Professor of SANKEN at Osaka University from 2021 to 2023. Since 2023, he has been a Designated Senior Assistant Professor with the Academy of Emerging Sciences, Chubu University. His research interest includes machine learning applications to biological and cognitive science.

MASAYUKI NUMAO (Member, IEEE) received the B.Eng. degree in electrical and electronics engineering and the Ph.D. degree in computer science from the Tokyo Institute of Technology, Japan, in 1982 and 1987, respectively. He was with the Department of Computer Science, Tokyo Institute of Technology, from 1987 to 2003; and a Visiting Scholar with CSLI, Stanford University, USA, from 1989 to 1990. He is currently a Professor with the Department of Architecture for Intelligence, The Institute of Scientific and Industrial Research (ISIR), Osaka University, Japan. His research interests include artificial intelligence, machine learning, affective computing, and empathic computing. He is a member of the Information Processing Society of Japan, JSAI, the Japanese Cognitive Science Society, the Japan Society for Software Science and Technology, and the American Association for Artificial Intelligence.

KEN-ICHI FUKUI (Member, IEEE) received the master's degree from Nagoya University, Japan, in 2003, and the Ph.D. degree in information science from Osaka University, Japan, in 2010. He has been an Associate Professor with SANKEN (The Institute of Scientific and Industrial Research), Osaka University, since 2015. He was a Specially Appointed Assistant Professor (2005–2010) and an Assistant Professor (2010–2015) with SANKEN. His research interests include machine learning, data mining, and its environmental contribution. He is a member of the IEEE Computer Society.