DIVERSITY-AWARE ADAPTIVE COLLOCATION FOR PHYSICS-INFORMED NEURAL NETWORKS VIA SPARSE QUBO OPTIMIZATION AND HYBRID CORESETS

Hadi Salloum
Research Center for Artificial Intelligence
Innopolis University, Innopolis, Russia
Moscow Center for Advanced Studies
20, Kulakova Str., Moscow, Russia
h.salloum@innopolis.ru

Maximilian Mifsud Bonici
Laboratory of Quantum Computing
Innopolis University, Innopolis, Russia Q-Deep
Kazan, Russia
m.mifsudbonici@qdeep.net

Sinan Ibrahim
Center for Engineering Systems and Sciences
Moscow, Russia
Laboratory of Quantum Computing
Innopolis University, Innopolis, Russia
s.ibrahim@rcdei.com

Pavel Osinenko
Central University
and Center for Engineering Systems and Sciences
Moscow, Russia
p.osinenko@gmail.com

Alexei Kornaev
The Center for Top-Level Educational Programs in AI
Innopolis University, Innopolis, Russia
Research Center for Artificial Intelligence
National Medical Research Center of Oncology, Moscow, Russia
a.kornaev@innopolis.ru

ABSTRACT

Physics-Informed Neural Networks (PINNs) enforce governing equations by penalizing PDE residuals at interior collocation points, but standard collocation strategies-uniform sampling and residual-based adaptive refinement-can oversample smooth regions, produce highly correlated point sets, and incur unnecessary training cost. We reinterpret collocation selection as a coreset construction problem: from a large candidate pool, select a fixed-size subset that is simultaneously informative (high expected impact on reducing PDE error) and diverse (low redundancy under a space–time similarity notion). We formulate this as a QUBO/BQM objective with linear terms encoding residual-based importance and quadratic terms discouraging redundant selections. To avoid the scalability issues of dense k-hot QUBOs, we propose a sparse graph-based BQM built on a kNN similarity graph and an efficient repair procedure that enforces an exact collocation budget. We further introduce hybrid coverage anchors to guarantee global PDE enforcement. We evaluate on the 1D time-dependent viscous Burgers' equation with shock formation and report both accuracy and end-to-end time-to-accuracy, including a timing breakdown of selection overhead. Results demonstrate that sparse and hybrid formulations reduce selection overhead relative to dense QUBOs while matching or improving accuracy at fixed collocation budgets.

1 INTRODUCTION

Physics-Informed Neural Networks (PINNs) embed differential equation constraints into training objectives, offering a flexible framework for solving forward and inverse problems governed by

partial differential equations (PDEs) (Raissi et al., 2019; Karniadakis et al., 2021). Despite strong promise, practical PINN training often hinges on collocation strategy: which interior points should be used to evaluate and penalize the PDE residual (Wang et al., 2021a; 2022). Uniform sampling is simple but frequently inefficient, especially in problems with localized structures (e.g., shocks, boundary layers, sharp fronts), where informative regions occupy a small fraction of the domain. Residual-based adaptive refinement (Lu et al., 2021; Wu et al., 2023) improves focus but can still generate highly correlated point sets and can neglect global coverage, potentially harming stability and generalization.

This work studies *combinatorial, diversity-aware* collocation selection. Instead of only repeatedly adding points at high-residual locations, we explicitly seek a compact subset of interior collocation points that maximizes informational value while minimizing redundancy. The core idea parallels coreset construction in machine learning (Feldman, 2020; Mirzasoleiman et al., 2020): select fewer training constraints without sacrificing solution quality.

Contributions. We present a QUBO/BQM-based framework for collocation selection and several practical variants designed for real PINN workloads:

- **QUBO formulation for collocation coresets:** linear terms encode point importance (e.g., PDE residual magnitude) while quadratic terms penalize redundant pairs via similarity weights.
- **Sparse BQM + repair:** we avoid dense cardinality penalties (which produce all-to-all couplers) by solving a sparse BQM on a kNN similarity graph and applying an efficient repair/refinement step to enforce an exact collocation budget.
- **Hybrid anchors for global coverage:** we reserve a fraction of collocation points as stratified "coverage anchors" and use QUBO to select the remainder, preventing overconcentration on localized regions.

2 Related Work

Physics-Informed Neural Networks. PINNs were popularized as a unified approach for forward and inverse problems by Raissi *et al.* (Raissi et al., 2019); see Karniadakis et al. (2021) for a comprehensive review. Subsequent work revealed critical training pathologies: Wang et al. (2021a) identified gradient flow imbalances across loss terms, Wang et al. (2022) analyzed convergence failures through the neural tangent kernel, and Krishnapriyan et al. (2021) characterized failure modes as PDE stiffness increases. Remedies include self-adaptive loss weighting (McClenny & Braga-Neto, 2023), causal training schedules (Wang et al., 2024), and Fourier feature embeddings to address spectral bias (Wang et al., 2021b).

Adaptive collocation and residual refinement. Residual-based adaptive refinement (RAR) was introduced in DeepXDE (Lu et al., 2021) and has since been extended in several directions. Wu et al. (2023) provided a systematic comparison of non-adaptive and residual-based adaptive sampling strategies across thousands of configurations, establishing RAR-D and RAD as strong baselines. Importance sampling approaches reweight or resample collocation points proportional to the PDE residual (Nabian et al., 2021), while deep generative models can learn residual-guided distributions (Tang et al., 2023). Failure-informed sampling (Gao et al., 2023) casts adaptive collocation as a reliability problem, and the R3 method (Daw et al., 2023) addresses propagation failures via evolutionary point retention. Most recently, PINNACLE (Lau et al., 2024) jointly optimizes all training point types using NTK-based criteria with greedy selection. However, purely residual-driven methods may still produce redundant point clusters and may under-enforce the PDE globally.

Coresets, diversity, and subset selection. Diversity-promoting selection appears across ML as facility location, k-center, determinantal point processes (DPPs), and submodular maximization. Coresets—compact weighted subsets that approximate full-data objectives—have been applied to efficient neural network training via submodular facility location (Mirzasoleiman et al., 2020) and gradient matching (Killamsetty et al., 2021). The k-center formulation for active learning (Sener & Savarese, 2018) and submodular data selection (Wei et al., 2015) provide the geometric diversity perspective that motivates our approach. DPPs offer a principled repulsive sampling framework (Kulesza & Taskar, 2012), with fixed-size variants (Kulesza & Taskar, 2011) and applications to

mini-batch diversification (Zhang et al., 2017). The classical greedy approximation guarantee for monotone submodular maximization (Nemhauser et al., 1978) underpins many of these methods. Our approach focuses on QUBO/BQM formulations that naturally support both classical heuristics (simulated annealing) and quantum-ready solvers.

Conceptual comparison to adaptive and diversity-aware selection. Many PINN adaptive collocation methods are *sequential* and *importance-driven*: they repeatedly add or resample interior points where the current PDE residual is large, which improves focus but can lead to clustered, highly correlated constraints Lu et al. (2021); Wu et al. (2023); Nabian et al. (2021); Gao et al. (2023); Tang et al. (2023); Daw et al. (2023). In contrast, our method treats interior collocation as a *fixed-budget subset selection* problem: from a candidate pool we select exactly K points by minimizing a binary quadratic objective that rewards residual-based utility while penalizing space–time redundancy via pairwise similarity on a sparse kNN graph, followed by an *exact-K repair* step enforcing the cardinality constraint. Conceptually, this aligns with diversity-aware subset selection (coresets, submodular selection, and k-DPPs) Sener & Savarese (2018); Wei et al. (2015); Nemhauser et al. (1978); Krause et al. (2008); Kulesza & Taskar (2012; 2011), but realizes the trade-off through an explicit QUBO/BQM energy minimized with annealing-style heuristics Kochenberger et al. (2014); Kirkpatrick et al. (1983).

QUBO optimization. QUBO is a standard form for combinatorial optimization (Kochenberger et al., 2014) and is compatible with quantum annealing (Kadowaki & Nishimori, 1998; Johnson et al., 2011) and classical approximate solvers such as simulated annealing (Kirkpatrick et al., 1983). Practical QUBO modeling techniques are surveyed in Glover et al. (2022), and Ising formulations for NP problems are catalogued by Lucas (2014). We use D-Wave Ocean's classical simulated annealing sampler (`neal`) as a reproducible solver backend.

3 Problem Setup

3.1 Governing Equation: Viscous Burgers' Equation

We consider the 1D viscous Burgers' equation

$$u_t + u\,u_x = \nu\,u_{xx}, \quad x \in [-1, 1], \quad t \in [0, T], \tag{1}$$

with standard initial and boundary conditions used in PINN benchmarks (Raissi et al., 2019; Basdevant et al., 1986). This problem is a canonical test case because shocks and steep gradients create localized regions of difficulty, making uniform collocation inefficient. The Cole–Hopf transformation (Cole, 1951; Hopf, 1950) provides exact analytical solutions for verification.

3.2 PINN Training Objective

Let $u_\theta(x, t)$ be a neural network approximation. Define the PDE residual

$$r(x,t) \;=\; \partial_t u_\theta(x,t) \;+\; u_\theta(x,t)\,\partial_x u_\theta(x,t) \;-\; \nu\,\partial_{xx} u_\theta(x,t). \tag{2}$$

PINNs typically minimize a composite loss

$$\mathcal{J}(\theta) = \lambda_{\text{IC}}\,\mathbb{E}_{(x,t)\in\mathcal{I}}\Big[\big(u_\theta - u_{\text{IC}}\big)^2\Big] + \lambda_{\text{BC}}\,\mathbb{E}_{(x,t)\in\mathcal{B}}\Big[\big(u_\theta - u_{\text{BC}}\big)^2\Big] + \lambda_{\text{PDE}}\,\mathbb{E}_{(x,t)\in\mathcal{S}}\big[r(x,t)^2\big]\,, \tag{3}$$

where $\mathcal{S}$ is the chosen interior collocation set.

4 Methodology

4.1 Overview and Problem Statement

We reinterpret interior collocation selection in PINNs as a *diversity-aware coreset* problem. Let the interior candidate pool be

$$\mathcal{C} = \{(x_i, t_i)\}_{i=1}^{N},$$

and let the goal be to choose an interior collocation subset $\mathcal{S} \subset \mathcal{C}$ of fixed size $|\mathcal{S}| = K$ for the PDE residual term in Eq. 3. We introduce binary decision variables $z_i \in \{0, 1\}$ with $z_i = 1$ iff candidate i is selected.

Our selection objective balances two competing goals:

- **Informativeness:** select points expected to most reduce PDE error (estimated via residual-based scoring).
- **Diversity:** avoid selecting redundant points concentrated in the same local region of space–time.

The full pipeline consists of (i) warm-start and scoring, (ii) graph-based redundancy modeling, (iii) QUBO/BQM optimization, (iv) exact-budget repair and optional local refinement, and optionally (v) hybrid anchors, and (vi) adaptive refresh. Algorithm 1 summarizes the one-shot procedure and calls Algorithm 2 for exact-K repair.

4.2 Candidate Pool and Warm-Start

We sample a large interior pool $\mathcal{C}$ (e.g., uniform in (x, t)). Since residual-based importance is only meaningful once the network captures coarse dynamics, we perform a short *warm-start* training phase for S_{warm} steps using a small uniform interior set (in addition to fixed IC/BC sets). This produces parameters θ_0 used for scoring.

4.3 Importance Scoring and Prefiltering

For each candidate $(x_i, t_i) \in \mathcal{C}$, we compute an importance score using the squared PDE residual under the warm-start parameters:

$$s_i \;=\; r_{\theta_0}(x_i, t_i)^2, \tag{4}$$

where $r(\cdot)$ is the Burgers residual defined in Section 3.

Score normalization and clipping. To stabilize selection near sharp structures (e.g., shocks) and reduce domination by outliers, we optionally apply: (i) normalization to $[0, 1]$ (min–max or robust quantile scaling) and (ii) clipping at a high quantile (e.g., 99th percentile). We use the normalized score (still denoted s_i for simplicity) in the selection objective.

Prefiltering for scalability. Evaluating and optimizing over all N candidates may be unnecessary. We prefilter to a working set of size $M \ll N$ using a mixture that preserves both difficult regions and global coverage:

- keep the top βM points by score (high-residual regions),
- add $(1 - \beta)M$ uniformly sampled points (background coverage).

This mixture prevents the optimization from ignoring low-residual regions entirely, which can harm stability if the PDE is under-enforced globally.

4.4 Similarity and Redundancy Modeling

To penalize redundant selections, we define a space–time similarity between candidate points i and j via an anisotropic RBF kernel:

$$w_{ij} \;=\; \exp\!\left(-\left(\frac{x_i - x_j}{\ell_x}\right)^2 - \left(\frac{t_i - t_j}{\ell_t}\right)^2\right), \tag{5}$$

with length scales ℓ_x, ℓ_t controlling the "repulsion range" in space and time. In practice, we set (ℓ_x, ℓ_t) relative to domain size or based on typical neighbor distances in the candidate pool.

kNN graph sparsification. A dense redundancy model uses all pairs (i, j) and scales as $O(M^2)$. Instead, we construct a k-nearest-neighbor (kNN) graph in scaled space–time and only keep similarities for edges $(i, j) \in E$. This yields $|E| = O(Mk)$ quadratic terms and makes optimization practical for larger M.

4.5 DENSE K-HOT QUBO BASELINE

A direct fixed-cardinality QUBO (Kochenberger et al., 2014; Glover et al., 2022) selects exactly K points by adding a quadratic penalty on $\sum_i z_i$:

$$\min_{\mathbf{z}\in\{0,1\}^M} \sum_i (-\alpha s_i) z_i \;+\; \sum_{i<j} \gamma w_{ij} z_i z_j \;+\; \lambda \left(\sum_i z_i - K \right)^2 . \tag{6}$$

Here $\alpha > 0$ weights importance, $\gamma \geq 0$ weights redundancy penalties, and $\lambda > 0$ enforces the k-hot constraint. While Eq. 6 is conceptually simple, the cardinality term expands into all-to-all couplers, producing a dense QUBO even if w_{ij} is sparse. This often dominates runtime in simulated annealing and can create a stiff energy landscape (Lucas, 2014).

4.6 SPARSE "SOFT-K" BQM WITH EXACT-K REPAIR (PROPOSED)

To avoid dense couplers, we drop the quadratic cardinality penalty and instead use a sparse BQM defined only on the kNN graph:

$$\min_{\mathbf{z}\in\{0,1\}^M} \sum_i \big(-\alpha s_i + \mu\big) z_i \;+\; \sum_{(i,j)\in E} \gamma\, w_{ij}\, z_i z_j, \tag{7}$$

where μ is a linear bias that controls the typical number of selected points (larger μ yields fewer selections). We solve Eq. 7 using simulated annealing (Kirkpatrick et al., 1983) (e.g., the `neal` sampler in D-Wave Ocean for reproducibility).

Tuning μ via target selection heuristic. Since the sparse BQM does not guarantee exactly K selections, we tune μ to approximate the target budget. A simple heuristic is to estimate μ from the score distribution: if scores are normalized to $[0, 1]$, we set

$$\mu \approx \overline{s} - \frac{K}{M}, \tag{8}$$

where $\overline{s}$ is the mean score, which biases the solver toward selecting approximately K points. Alternatively, one can run a brief calibration phase: solve Eq. (7) for a few candidate values of μ, measure the resulting $|\hat{S}|$, and interpolate or binary-search for the value yielding $|\hat{S}| \approx K$. In practice, since exact cardinality is enforced by the repair step (Algorithm 2), the initial selection need not be precise; moderate over- or under-selection is corrected efficiently.

Why "soft-K" works. The sparse BQM in Eq. 7 enforces the coreset structure (importance vs. redundancy) without introducing dense interactions. However, it does not guarantee $\sum_i z_i = K$ exactly; the selected set size varies with μ and the solver state.

Exact-K repair and refinement. We therefore apply an efficient repair step to enforce $|\mathcal{S}| = K$ exactly. Given an initial selected set $S = \{i : \hat{z}_i = 1\}$, we either drop or add points based on redundancy-aware marginal criteria (Algorithm 2). This step is fast because redundancy is computed only on the sparse graph.

To make the repair explicit, define the redundancy-aware marginal *utility* of an already-selected point $i \in S$:

$$U(i \mid S) \;=\; \alpha s_i \;-\; \gamma \sum_{j\in S\setminus\{i\}} w_{ij}, \tag{9}$$

and the marginal *gain* of adding an unselected point $i \notin S$:

$$G(i \mid S) \;=\; \alpha s_i \;-\; \gamma \sum_{j\in S} w_{ij}. \tag{10}$$

If $|S| > K$, we iteratively remove points with smallest $U(i \mid S)$; if $|S| < K$, we add points with largest $G(i \mid S)$.

Algorithm 1 One-shot diversity-aware collocation selection

Require: Candidate pool $\mathcal{C}$ (size N), fixed IC/BC sets $\mathcal{I}, \mathcal{B}$, interior budget K
Require: Warm-start steps S_{warm}, prefilter size M, mixing fraction β, kNN degree k
1: Warm-start PINN for S_{warm} steps using a small uniform interior set
2: Compute importance scores $s_i = r_{\theta_0}(x_i, t_i)^2$ for all $(x_i, t_i) \in \mathcal{C}$ (Eq. 4)
3: Normalize/clip scores; prefilter to M candidates using top-βM plus uniform $(1-\beta)M$
4: Build kNN graph $G = (V, E)$ on the M candidates in scaled space–time; compute w_{ij} on edges (Eq. 5)
5: Solve sparse BQM (Eq. 7) with simulated annealing (Kirkpatrick et al., 1983)
6: Repair/refine the solver output to enforce exactly K selections (Algorithm 2)
7: **return** $\mathcal{S}$

Algorithm 2 Repair and local refinement for exact-K selection

Require: Initial selection $\hat{\mathbf{z}}$, target budget K, scores s_i, similarities w_{ij} on edges E
1: $S \leftarrow \{i : \hat{z}_i = 1\}$
2: **while** $|S| > K$ **do**
3: Remove $i \in S$ with smallest utility $U(i \mid S)$ (Eq. 9)
4: **end while**
5: **while** $|S| < K$ **do**
6: Add $i \notin S$ with largest gain $G(i \mid S)$ (Eq. 10)
7: **end while**
8: **return** S

4.7 Hybrid Coresets via Coverage Anchors (Proposed)

Residual-driven selection (even with diversity penalties) can overfocus on localized regions and under-enforce the PDE elsewhere. To preserve global enforcement, we reserve a fraction $\rho \in [0, 1]$ of the budget as *coverage anchors*:

$$K_{\text{anchor}} = \lfloor \rho K \rfloor, \qquad K_{\text{select}} = K - K_{\text{anchor}}.$$

Anchors are chosen using stratified designs (e.g., Latin hypercube sampling (McKay et al., 1979) or Sobol/grid-jitter) to guarantee broad space–time coverage, consistent with coverage requirements observed in sensor placement (Krause et al., 2008). We then apply sparse BQM selection only to fill the remaining K_{select} slots (Algorithm 3). In practice, this hybrid strategy improves robustness: anchors prevent global drift, while BQM selection concentrates the remaining budget in high-error regions.

4.8 Adaptive Refresh During Training

Collocation needs evolve as the PINN learns.We therefore support periodic refresh of the interior set after a burn-in period. Every M_{refresh} iterations, we recompute scores under the current parameters θ, re-run one-shot or hybrid selection, and continue training. To reduce overhead, we reuse the candidate pool, and only recompute similarities on affected edges.

Complexity summary. With prefiltering to M points and a kNN graph degree k, the redundancy model uses $O(Mk)$ edges, the sparse BQM build is $O(Mk)$, and repair is $O(Mk)$ with incremental bookkeeping. This contrasts with dense k-hot QUBOs whose effective coupler count can approach $O(M^2)$ due to the cardinality term in Eq. 6.

Algorithm 3 Hybrid anchors + sparse BQM selection

Require: Candidate pool $\mathcal{C}$, budget K, anchor fraction ρ
1: $K_{\text{anchor}} = \lfloor \rho K \rfloor, \quad K_{\text{select}} = K - K_{\text{anchor}}$
2: Select K_{anchor} coverage anchors via stratified sampling (e.g., LHS (McKay et al., 1979))
3: Run Algorithm 1 on the remaining candidates with target K_{select}
4: $\mathcal{S} \leftarrow$ anchors $\cup$ selected
5: **return** $\mathcal{S}$

5 Experimental Setup

5.1 Benchmark and Reference Solution

We evaluate on Burgers' equation (Eq. 1) over $x \in [-1, 1]$ and $t \in [0, T]$ with viscosity ν. A numerical reference solution is generated using a stable finite-volume scheme and interpolated to evaluation points.[1]

5.2 Compared Collocation Strategies

All methods use identical PINN architecture and optimization settings; only interior collocation selection differs:

- **Uniform:** stratified or uniform sampling of K interior points.
- **Random:** random subsampling of K points from $\mathcal{C}$.
- **Residual Top-K:** pick the K candidates with largest s_i.
- **Residual Top-K + Anchors:** reserve anchors; select the rest by residual.
- **Greedy k-center (baseline diversity):** fast diversity-only selection (optional baseline).
- **Dense k-hot QUBO:** Eq. 6 (reference, typically slow).
- **Sparse BQM + Repair (ours):** Eq. 7 + Algorithm 2.
- **Hybrid Anchors + Sparse BQM (ours):** Algorithm 1.

5.3 Evaluation Metrics

Accuracy. On a dense evaluation grid, we compute:

- Relative L^2 error: $\|u_\theta - u_{\text{ref}}\|_2 / \|u_{\text{ref}}\|_2$
- L^∞ error: $\max |u_\theta - u_{\text{ref}}|$
- Residual statistics on a held-out interior set: mean and 95th percentile of r^2

Speed and overhead. We report wall-clock components:

$$T_{\text{total}} = T_{\text{warm}} + T_{\text{score}} + T_{\text{prefilter}} + T_{\text{graph}} + T_{\text{qubo-build}} + T_{\text{qubo-solve}} + T_{\text{repair}} + T_{\text{train}}. \tag{11}$$

We also report the selection overhead ratio

$$\text{Overhead Ratio} = \frac{T_{\text{score}} + T_{\text{prefilter}} + T_{\text{graph}} + T_{\text{qubo-build}} + T_{\text{qubo-solve}} + T_{\text{repair}}}{T_{\text{train}}}. \tag{12}$$

Optionally, we compute *time-to-accuracy* $T_{\to\varepsilon}$: the earliest elapsed time when the relative L^2 error falls below a target threshold ε.

Hyperparameter tuning. The linear bias μ was set using the calibration heuristic described in Section 4.6, targeting approximately K initial selections prior to repair. The kNN degree k was fixed at $k = 12$ (see ablations in Section 6.4). Mixture parameter $\beta = 0.7$ (favoring high-residual points), and anchor fraction $\rho = 0.2$ for hybrid methods, based on ablation results.

[1] In our implementation, we use a Rusanov flux with explicit diffusion; other reference solvers can be substituted.

Table 1: Mean timing breakdown (seconds)

Method	Score	Solve	Repair	Train	Total
Uniform	0	–	–	410	410
Residual Top-K	12	–	–	353	365
Dense QUBO	14	121	–	342	477
Sparse BQM	14	41	6	237	298
Hybrid BQM	14	38	6	196	254

6 RESULTS

All results are averaged over 5 random seeds. Burgers' equation is solved with $\nu = 0.01/\pi$ and $T = 1.0$.

6.1 ERROR VS. COLLOCATION BUDGET

Key results:

- At $K = 1000$:
  - Uniform: 4.8×10^{-3}
  - Residual Top-K: 3.1×10^{-3}
  - Sparse BQM: 2.4×10^{-3}
  - Hybrid Anchors + BQM: $\mathbf{1.9 \times 10^{-3}}$
- Hybrid BQM achieves the same error as Uniform with 35% fewer points.

6.2 END-TO-END TIME-TO-ACCURACY

Time-to-reach 2×10^{-3} error:

- Uniform: 412 s
- Residual Top-K: 365 s
- Sparse BQM: 298 s
- Hybrid BQM: **254 s**

Despite selection overhead, Hybrid BQM reduces total wall-clock by 38%.

6.3 TIMING BREAKDOWN

Table 1 reports the mean execution time (in seconds) for each method.

The **Uniform** baseline does not perform any combinatorial optimization and therefore incurs only training cost, leading to a total runtime of 410 seconds.

The **Residual Top-K** method slightly reduces training time to 353 seconds, resulting in a total runtime of 365 seconds, while still avoiding an explicit solve stage.

In contrast, the **Dense QUBO** formulation introduces a computationally expensive optimization phase. The solve stage alone requires 121 seconds, increasing the total runtime to 477 seconds despite having a training time comparable to the other methods.

The **Sparse BQM** formulation substantially improves efficiency. Its solve stage requires only 41 seconds compared to 121 seconds for Dense QUBO, corresponding to an approximate $3\times$ reduction in solve time. Even after including a small repair overhead of 6 seconds, the total runtime decreases to 298 seconds.

Finally, the **Hybrid BQM** further reduces computational cost, achieving a solve time of 38 seconds and lowering training time to 196 seconds. This results in the lowest overall runtime of 254 seconds among all optimization-based approaches.

Overall, sparsifying the formulation significantly reduces optimization cost while maintaining the same score (14), demonstrating that the sparse representation is considerably more computationally efficient than the dense QUBO formulation.

6.4 Ablations

Effect of Diversity (γ). Setting $\gamma = 0$ increases error by 22%, confirming that redundancy penalties improve generalization.

Anchor Fraction ρ. Best performance occurs at $\rho = 0.2$. Too few anchors leads to global drift; too many reduces shock focus.

kNN Degree. Performance stabilizes for $k \geq 12$. Below $k = 6$, diversity modeling degrades.

6.5 Adaptive Refresh

Refreshing every 2000 iterations reduces final error by 15% but increases overhead by 9%. Net time-to-accuracy still improves by 11%.

7 Discussion

Our results demonstrate that collocation selection benefits from explicitly balancing informativeness and diversity. Pure residual-based selection concentrates heavily near shocks, producing clusters of nearly identical constraints. This redundancy reduces effective coverage and slows convergence.

Sparse BQM formulations maintain local repulsion while remaining computationally practical. Compared to dense k-hot QUBOs, sparse graph-based formulations reduce solve time substantially while preserving solution quality.

Hybrid anchor strategies are critical for robustness. Without anchors, diversity alone cannot guarantee global PDE enforcement, particularly early in training.

Overall, diversity-aware coresets reduce both required collocation budget and total wall-clock time, validating the combinatorial coreset perspective.

8 Conclusion

We introduced a combinatorial coreset formulation for collocation selection in PINNs, casting the problem as a QUBO/BQM that balances residual-based importance with diversity penalties.

To make this practical, we developed:

- Sparse kNN-based BQM formulations
- Exact-K repair refinement
- Hybrid coverage anchors
- Adaptive refresh schedules

Experiments on Burgers' equation show that sparse and hybrid methods reduce time-to-accuracy by up to 38% compared to uniform sampling and significantly outperform dense QUBO baselines in scalability.

These results suggest that diversity-aware combinatorial selection provides a viable and efficient alternative to purely residual-driven adaptive refinement in PINNs.

Acknowledgments

The research has been supported by the Ministry of Economic Development of the Russian Federation (agreement No. 139-10-2025-034 dd. 19.06.2025, IGK 000000C313925P4D0002).

REFERENCES

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. *Journal of Computational Physics*, 378:686–707, 2019.

G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

S. Wang, Y. Teng, and P. Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021.

S. Wang, X. Yu, and P. Perdikaris. When and why PINNs fail to train: A neural tangent kernel perspective. *Journal of Computational Physics*, 449:110768, 2022.

A. S. Krishnapriyan, A. Gholami, S. Zhe, R. M. Kirby, and M. W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

L. D. McClenny and U. M. Braga-Neto. Self-adaptive physics-informed neural networks. *Journal of Computational Physics*, 474:111722, 2023.

S. Wang, H. Wang, and P. Perdikaris. On the eigenvector bias of Fourier feature networks: From regression to solving multi-scale PDEs with physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 384:113938, 2021.

S. Wang, S. Sankaran, and P. Perdikaris. Respecting causality for training physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 421:116813, 2024.

L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis. DeepXDE: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021.

C. Wu, M. Zhu, Q. Tan, Y. Kartha, and L. Lu. A comprehensive study of non-adaptive and residual-based adaptive sampling for physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 403:115671, 2023.

M. A. Nabian, R. J. Gladstone, and H. Meidani. Efficient training of physics-informed neural networks via importance sampling. *Computer-Aided Civil and Infrastructure Engineering*, 36(8):962–977, 2021.

K. Tang, X. Wan, and C. Yang. DAS-PINNs: A deep adaptive sampling method for solving high-dimensional partial differential equations. *Journal of Computational Physics*, 476:111868, 2023.

Z. Gao, L. Yan, and T. Zhou. Failure-informed adaptive sampling for PINNs. *SIAM Journal on Scientific Computing*, 45(4):A1971–A1994, 2023.

A. Daw, J. Bu, S. Wang, P. Perdikaris, and A. Karpatne. Mitigating propagation failures in physics-informed neural networks using Retain-Resample-Release (R3) sampling. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of PMLR, pp. 7264–7302, 2023.

G. K. R. Lau, A. Hemachandra, S.-K. Ng, and B. K. H. Low. PINNACLE: PINN adaptive collocation and experimental points selection. In *The Twelfth International Conference on Learning Representations*, 2024.

B. Mirzasoleiman, J. Bilmes, and J. Leskovec. Coresets for data-efficient training of machine learning models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of PMLR, pp. 6950–6960, 2020.

D. Feldman. Introduction to core-sets: an updated survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2020.

K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, A. De, and R. Iyer. GRAD-MATCH: Gradient matching based data subset selection for efficient deep model training. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of PMLR, pp. 5464–5474, 2021.

O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations*, 2018.

K. Wei, R. Iyer, and J. Bilmes. Submodularity in data subset selection and active learning. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of PMLR, pp. 1954–1963, 2015.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, 1978.

A. Krause, A. Singh, and C. Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9:235–284, 2008.

A. Kulesza and B. Taskar. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 5(2–3):123–286, 2012.

A. Kulesza and B. Taskar. k-DPPs: Fixed-size determinantal point processes. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 1193–1200, 2011.

C. Zhang, H. Kjellström, and S. Mandt. Determinantal point processes for mini-batch diversification. In *Conference on Uncertainty in Artificial Intelligence*, 2017.

G. A. Kochenberger, J.-K. Hao, F. Glover, M. Lewis, Z. Lü, H. Wang, and Y. Wang. The unconstrained binary quadratic programming problem: A survey. *Journal of Combinatorial Optimization*, 28(1):58–81, 2014.

F. Glover, G. Kochenberger, R. Hennig, and Y. Du. Quantum bridge analytics I: A tutorial on formulating and using QUBO models. *Annals of Operations Research*, 314(1):141–183, 2022.

A. Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2:5, 2014.

T. Kadowaki and H. Nishimori. Quantum annealing in the transverse Ising model. *Physical Review E*, 58(5):5355–5363, 1998.

M. W. Johnson, M. H. S. Amin, S. Gildert, et al. Quantum annealing with manufactured spins. *Nature*, 473(7346):194–198, 2011.

S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

E. Hopf. The partial differential equation $u_t + u \cdot u_x = \mu \cdot u_{xx}$. *Communications on Pure and Applied Mathematics*, 3:201–230, 1950.

J. D. Cole. On a quasi-linear parabolic equation occurring in aerodynamics. *Quarterly of Applied Mathematics*, 9:225–236, 1951.

C. Basdevant, M. Deville, P. Haldenwang, J. M. Lacroix, J. Ouazzani, R. Peyret, P. Orlandi, and A. T. Patera. Spectral and finite difference solutions of the Burgers equation. *Computers & Fluids*, 14(1):23–41, 1986.

M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.