

# Сравнительный анализ методов PINN и PIKAN при моделировании одномерного распределения температуры в металлах ( $\text{VO}_2$ и Cu)

М. А. Семенченко

Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И. Ульянова (Ленина)

i.am.mikhail.semenchenko@gmail.com

**Аннотация.** В работе представлен сравнительный анализ методов научного машинного обучения (SciML) – нейросетей, информированных физикой (PINN), и их альтернативы на базе представления Колмогорова–Арнольда (PIKAN) – при решении прямой и обратной задач нестационарной теплопроводности. Рассмотрены ограничения традиционных численных методов (МКР, МКЭ), связанные с условием устойчивости Куранта–Фридрихса–Леви. Приведено математическое обоснование перехода от обработки сигнала в узлах к обучению на рёбрах графа на основе теоремы суперпозиции, являющейся решением 13-й проблемы Гильберта. Описана программная реализация на базе библиотек PyTorch и DeepXDE. Результаты экспериментов на меди (Cu) и диоксиде ванадия ( $\text{VO}_2$ ) подтверждают сходимость нейросетевых решений и преимущество PIKAN в части локальной адаптивности к фазовым переходам, а также более высокую устойчивость при восстановлении коэффициента температуропроводности по зашумлённым данным.

**Ключевые слова:** SciML; PINN; PIKAN; DeepXDE; диоксид ванадия; теплопроводность; сплайны; условие CFL; автоматическое дифференцирование; спектральное смещение; обратные задачи, теорема Колмогорова–Арнольда

## I. ВВЕДЕНИЕ

Решение дифференциальных уравнений в частных производных (ДУЧП), описывающих нестационарную теплопроводность, традиционно опирается на сеточные методы: метод конечных разностей (МКР) и метод конечных элементов (МКЭ). Несмотря на математическую строгость, применение явных схем интегрирования ограничено условием Куранта–Фридрихса–Леви (CFL), требующим существенного измельчения шага по времени при использовании мелкой пространственной сетки. Это приводит к экспоненциальному росту вычислительных затрат при моделировании высокоградиентных процессов, например, фазовых переходов в материалах [1, 2].

В последние годы парадигма научного машинного обучения (SciML) предложила бессеточный подход – физически-информированные нейронные сети (PINN) [3]. Интегрируя физические законы непосредственно в функцию потерь через автоматическое дифференцирование, PINN позволяют находить решения ДУЧП без привязки к расчётной сетке. Однако классические PINN на базе многослойного перцептрона

(MLP) подвержены проблеме «спектрального смещения» (spectral bias) – тенденции к преимущественному обучению низкочастотным компонентам решения [4]. Это затрудняет аппроксимацию резких градиентов, характерных для фазовых переходов в диоксиде ванадия ( $\text{VO}_2$ ).

Эффективной альтернативой выступают сети Колмогорова–Арнольда (KAN) [5], в которых обучаемые функции (например, B-сплайны) располагаются на рёбрах графа, а узлы выполняют простое суммирование. Применение таких сетей в задачах с физической информацией (PIKAN) [6, 7] обеспечивает локальную адаптивность и потенциально улучшенную точность при меньшем числе параметров.

В данной работе проводится систематическое сравнение PINN и PIKAN для одномерного уравнения теплопроводности на примере двух материалов: меди (Cu) с постоянными теплофизическими свойствами, и диоксида ванадия ( $\text{VO}_2$ ), испытывающего фазовый переход «диэлектрик-металл» при температуре 341 К (68°C). Анализируются свойства моделей при обучении на основе решений прямой задачи численными методами, эффективность использования их параметров, а также устойчивость PINN и PIKAN при решении обратной задачи идентификации коэффициента температуропроводности по зашумлённым измерениям. Помимо этого изложен опыт проведения вычислений на различных конфигурациях оборудования.

## II. МАТЕМАТИЧЕСКАЯ ПОСТАНОВКА ЗАДАЧИ

### A. Основное уравнение

Рассматривается одномерное уравнение нестационарной теплопроводности (уравнение Фурье):

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}, \quad \alpha = \frac{k}{\rho c_p}, \quad (1)$$

где  $T(x, t)$  – температура,  $\rho$  – плотность,  $c_p$  – удельная теплоёмкость,  $k$  – коэффициент теплопроводности,  $\alpha$  – коэффициент температуропроводности.

Для меди (Cu) теплофизические свойства полагаются постоянными:  $\alpha \approx 1.14 \times 10^{-4} \text{ м}^2/\text{с}$ . Для диоксида ванадия ( $\text{VO}_2$ ) в окрестности температуры фазового перехода  $T_c \approx 341 \text{ К}$  наблюдается скачкообразное изменение  $k$  и  $c_p$ : теплопроводность возрастает примерно на 60%, а теплоёмкость имеет резкий пик [11, 12]. В

результате уравнение (1) становится существенно нелинейным.

Моделирование диоксида ванадия осложнено наличием фазового перехода, сопровождающегося скачком коэффициента температуропроводности. Для обеспечения работы механизмов автоматического дифференцирования в SciML-моделях разрывная функция скачка значений температуры заменена на непрерывную аппроксимацию на базе гиперболического тангенса (2). Это позволяет сохранить высокую крутизну перехода в окрестности критической температуры, сохраняя при этом дифференцируемость вычислительного графа, что является обязательным требованием для градиентных методов оптимизации.

$$\alpha = \alpha \dots \dots \dots (2)$$

где  $\tau = tgh(10 \cdot (T - 0.5))$  – приближение скачка теплопроводности бесконечно дифференцируемой гладкой функцией (гиперболическим тангенсом).

### В. Прямая задача

В прямой задаче коэффициент  $\alpha$  и начально-краевые условия известны. Требуется найти температурное поле  $T(x, t)$  в области  $(x, t) \in [0, L] \times [0, t_{max}]$ . Начальное условие:  $T(x, 0) = \sin\left(\frac{\pi x}{L}\right)$ ; граничные условия Дирихле:  $T(0, t) = T(L, t) = 0$ . Нейросетевое приближение  $T_\theta(x, t)$  ищется минимизацией составной функции потерь:

$$L = L_{PDE} + L_{BC} + L_{IC}, (3)$$

где  $L_{PDE} = \frac{1}{N_f} \sum_{i=1}^{N_f} \left| \frac{\partial T_\theta}{\partial t}(x_f^i, t_f^i) - \alpha \frac{\partial^2 T_\theta}{\partial x^2}(x_f^i, t_f^i) \right|^2$ ,  $L_{BC}$  и  $L_{IC}$  – среднеквадратичные ошибки на граничных и начальных точках.

Физически однородные граничные условия первого рода (условия Дирихле) соответствуют закреплению концов стержня в массивных термостатах, поддерживающих постоянную нулевую температуру. В контексте обучения нейросетевых моделей наложение жёстких ограничений на границах расчётной области необходимо для выделения единственного решения из множества функций, удовлетворяющих исходному дифференциальному уравнению, что обеспечивает общую сходимость алгоритма.

Начальное условие определяет состояние системы в момент запуска процесса. Выбор синусоидального профиля позволяет моделировать плавное распределение тепловой энергии вдоль стержня с пиком в центре. В программной реализации это условие формирует компоненту функции потерь, вынуждая нейросетевую аппроксимацию соответствовать заданному вектору состояния в начальной точке времени.

### С. Обратная задача

В обратной задаче коэффициент  $\alpha$  неизвестен и подлежит определению по дополнительным измерениям температуры в отдельных точках. Наблюдения  $\hat{T}_j$  в точках  $(x_j, t_j)$  зашумлены:

$$\hat{T}_j = T(x_j, t_j) + \varepsilon_j (4)$$

Функция потерь (3) дополняется слагаемым, отвечающим за невязку с данными:

$$L = L_{PDE} + L_{BC} + L_{IC} + L_{data}, (5)$$

где  $L_{data} = \frac{1}{N_d} \sum_{j=1}^{N_d} |T_\theta(x_j, t_j) - \hat{T}_j|^2$ . Параметр  $\alpha$  включается в число обучаемых переменных наряду с весами сети.

### D. Архитектуры нейронных сетей

#### a. Классическая модель PINN (MLP)

В PINN решение аппроксимируется многослойным персептроном:

$$T_\theta(x, t) = N(x, t; \theta) (6)$$

где  $\theta$  – веса и смещения. Выход каждого нейрона:

$$h_i^{(l+1)} = \sigma\left(\sum_j W_{ij}^{(l)} h_j^{(l)} + b_i^{(l)}\right) (7)$$

с функцией активации  $\sigma = tg(h)$ , обеспечивающей гладкость, необходимую для вычисления производных [3].

#### b. Архитектура PIKAN (KAN)

PIKAN базируется на теореме Колмогорова–Арнольда о представлении многомерных функций суммами одномерных [13]:

$$f(x_1, \dots, x_n) = \sum_{q=1}^{2n+1} \Phi_q\left(\sum_{p=1}^n \phi_{q,p}(x_p)\right) (8)$$

В сети Колмогорова–Арнольда (KAN) обучаемые одномерные функции реализуются в виде B-сплайнов на рёбрах графа, а узлы выполняют простое суммирование [5]. Входной слой принимает координаты  $(x, t)$ , далее следуют скрытые слои с обучаемыми сплайнами, выходной слой выдаёт температуру. Прямое распространение сигнала в KAN-слое:

$$x_{l+1,j} = \sum_{i=1}^{n_l} \phi_{l,j,i}(x_{l,i}), (9)$$

где  $\phi_{l,j,i}$  – сплайн с обучаемыми коэффициентами.

Такая архитектура позволяет модели локально адаптироваться в зонах резких градиентов, что критически важно для такого материала как VO<sub>2</sub>.

#### c. Условие устойчивости Куранта (CFL)

Традиционные явные конечно-разностные схемы (МКР) жёстко ограничены условием Куранта–Фридрихса–Леви. Нарушение данного условия ведёт к численной неустойчивости и росту ошибок. Переход к методам SciML позволяет преодолеть это ограничение, так как нейросетевые модели обучаются на непрерывном пространстве-времени и не привязаны к жёсткой сетке, что открывает возможности для моделирования быстрых процессов без избыточных вычислительных затрат.

В рамках этой работы, формула, описывающая данное условие, выражается как:

$$\Delta t \leq \frac{\Delta x^2}{2\alpha} (10)$$

где  $\alpha$  – коэффициент температуропроводности из формулы (1).

### Е. Численный эталон (FDM)

Ввиду отсутствия прямых экспериментальных данных, в качестве контрольного решения, эталона, использована конечно-разностная аппроксимация. Выбор консервативного шага по времени ( $0.4 \times CFL$ ) гарантирует отсутствие вычислительных артефактов. Таким образом формируется верифицированная расчётная база, позволяющая оценивать погрешность новых архитектур PINN и PIKAN в контролируемых условиях.

В виде формулы её можно представить как:

$$t_i^{(n+1)} \leq t_i^n + \alpha \frac{\Delta t}{\Delta x^2} (t_{i+1}^n - 2t_i^n + t_{i-1}^n) \quad (11)$$

### III. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ И МЕТОДОЛОГИЯ

Моделирование выполнено на языке Python с использованием библиотек PyTorch, PyKAN и DeepXDE [14]. DeepXDE предоставляет высокоуровневые абстракции для задания геометрии, граничных условий и автоматического дифференцирования. Также была использована библиотека Python PyKAN, реализующая архитектуру PIKAN на практике.

В рамках модели геометрия задана на стержне длиной  $L = 1.0$  на временном интервале  $T_{max} = 1.0$ . Так как распределение температуры задано синусоидой, а концы стержня по условию модели закреплены в охлаждающих держателях, на двумерном графике зависимости температуры по длине стержня от времени в начальной точке ожидается увидеть градиент нагрева из центра стержня. На основе полученных данных и моделей также были сгенерированы карты распределения тепла при решении обратной задачи, а также карты сравнения ошибок PINN и PIKAN относительно эталона FDM.

Рассматриваются два случая:

1. Для меди устанавливается среда с коэффициентом температуропроводности  $\alpha = 0.5$  – в интервале от 0 до 1 – это результат нормирования реальной физической константы для удобства вычислений на ЭВМ.
2. Для диоксида ванадия этот же коэффициент задаётся динамически по закону гиперболического тангенса (2) для моделирования фазового перехода.

На основе расчётов численными методами проводится последовательное обучение модели с целью улучшить её предсказания о поведении материала.

Так как в реальных задачах у датчиков замеров могут быть помехи и ошибки, вводится зашумление входных данных (4), коими являются показания температуры материала в случайных точках, полученных после вычислений численными методами для верификации.

В обоих случаях целью обучения перед моделью ставится уменьшение ошибки в предсказаниях. В рамках этой работы были обучены модель PINN с архитектурой  $[2, 32, 32, 1]$  – примерно 1185 параметров, а также PIKAN с шириной  $[2, 10, 1]$ , сеткой  $grid = 5$  и  $k = 3$  – примерно 270 параметров. Так как библиотеки для расчётов В-сплайнов на GPU найдены не были, их вычисление производилось на CPU, что привело к серьёзному увеличению времени обучения, в связи с чем было принято решение уменьшить число параметров для

PIKAN. В соответствующем разделе статьи есть обзор пусков обучения с разными параметрами.

Таким образом, важно помнить, что у модели на базе PIKAN в данной работе примерно в 4 раза меньше число параметров, чем у модели на архитектуре PINN.

#### А. Метод дискретизации

Для формирования расчетной области и обучения моделей использовалась комбинированная стратегия распределения узлов. Внутри области  $[0, L] \times [0, T]$  генерировалось  $N_f = 1000$  точек для вычисления невязки уравнения; на границах  $N_{bc} = 200$  точек; на начальном условии  $N_{ic} = 200$  точек. Из численного эталона извлекалось 300 случайных точек («датчиков»), в которых фиксировались зашумленные значения температуры.

#### В. Оптимизация

Использован двухэтапный протокол оптимизации:

1. Adam (learning rate  $10^{-3}$ , 2000 итераций) – для выхода в окрестность глобального минимума.
2. L-BFGS (квазиньютоновский метод, tolerance  $10^{-8}$ ) – для финальной доводки решения до высокой точности [15, 16, 17].

Для данного этапа рекомендуется использовать вычислительные мощности GPU с целью ускорить процесс сходимости к ожидаемым значениям.

#### С. Псевдокод

Процесс обучения был запущен несколько раз с разными параметрами модели PINN и PIKAN. Были протестированы зависимости между числом параметров модели, времени тренировки и точностью сходимости для разных архитектур (PINN, PIKAN), материалов ( $Cu, VO_2$ ), а также устройств, на которых происходили вычисления (CPU, GPU). С полными данными можно ознакомиться в репозитории проекта на GitHub [19] – в рамках данной работы будет приведена выдержка в виде графиков, построенных на основе данных, структурированных в формате JSON.

Псевдокод (Алгоритм) определения сети и обучения в DeepXDE

1. Инициализация
  - 1.1. Задать параметры области:  $L = 1.0, T_{max} = 1.0$ .
  - 1.2. Установить уровни шума  $\eta \in \{0, 5, 15\}\%$ .
  - 1.3. Задать начальное приближение  $\alpha_{var} = 1.1$ .
2. Подготовка данных (FDM)
  - 2.1. Рассчитать эталонное решение  $t_{fdm}$  методом конечных разностей.
  - 2.2. Для  $VO_2$  применить модель фазового перехода  $\alpha(t)$ .
  - 2.3. Выбрать 300 случайных точек и добавить шум.
3. Конфигурация нейросетей
  - 3.1. PINN: Полносвязная сеть  $[2, 32, 32, 1]$ , активация  $\tanh$ .
  - 3.2. PIKAN: Сеть Колмогорова–Арнольда  $[2, 10, 1]$ , сетка 5, сплайны 3-го порядка.
  - 3.3. Сформулировать функцию потерь  $L_{total}$

#### 4. Оптимизация

- 4.1. Предварительное обучение: 2000 итераций алгоритмом Adam.
- 4.2. Финальная сходимость: оптимизатор L-BFGS.
- 4.3. Обработка данных: ошибок и параметра  $\alpha_{var}$ .

#### 5. Валидация

- 5.1. Восстановить полное поле температур  $t_{pred}$ .
- 5.2. Рассчитать относительную ошибку  $L^2$ .
- 5.3. Сравнить точность идентификации  $\alpha_{var}$  для PINN и PIKAN.

Псевдокод выше описывает скрипт, который произвёл модели способные произвести данные для карт ошибки и распределения температуры в этой работе. Далее этот скрипт будет именоваться “контрольным”. Контрольное обучение происходило полностью на CPU.

#### D. Длительность обучения



Рис. 1. Время, затраченное на этапы Adam и L-BFGS

Этап обучения модели разбит на два этапа: грубого поиска примерных значений посредством алгоритма Adam (2000 итераций), и последующего уточнения полученных параметров через L-BFGS (число шагов разнилось). На рис. 1 представлено отношение времени, затраченного на каждый из алгоритмов в рамках одного из запусков скрипта.

### IV. ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

#### A. Измеримые зависимости в процессе обучения

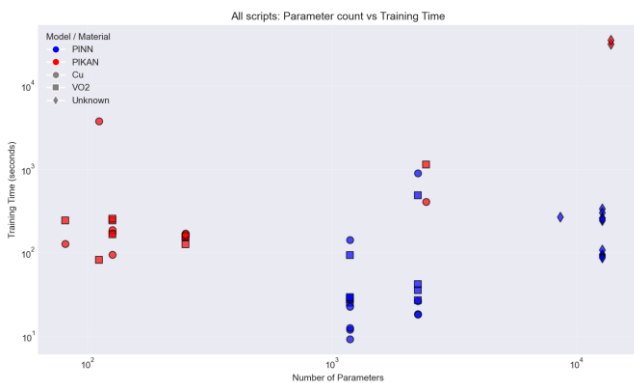


Рис. 2. Зависимость времени обучения от числа параметров

Рассмотрим зависимость между числом параметров модели и временем обучения среди всех запусков в рамках этой работы, представленную на рис. 2.

Обнаружено, что PIKAN обучался дольше при меньшем числе параметров, чем у PINN. Проверим возможно ли выявить иные зависимости.

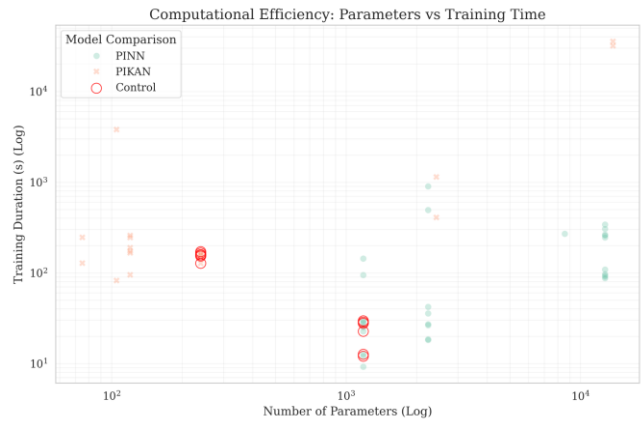


Рис. 3. Время обучения с отметками контрольного пуска

Можно заметить, что разброс времени обучения у PIKAN ниже, чем у PINN, для обоих материалов. Это означает, что оптимизация вычислений для PIKAN позволит улучшить показатели агностичны к наличию у материала фазовых переходов, что делает PIKAN перспективным подходом как универсальной архитектуры для материалов как с фазовым переходом, так и без.

Проверим как число параметров повлияло на итоговую сходимость.

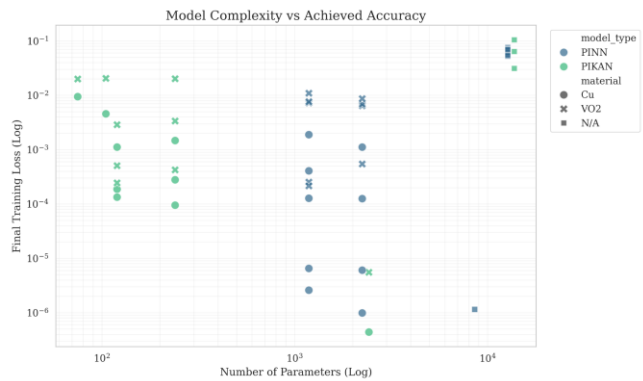


Рис. 4. Зависимость точности от числа параметров

Обе архитектуры демонстрируют схожий уровень точности, причём для PIKAN такой результат достигнут с меньшим числом параметров. Заметим, что на равном числе параметров, PIKAN продемонстрировал лучшую точность для обоих материалов в сравнении с PINN. В дальнейших работах имеет смысл проверить будет ли сохраняться эта тенденция на разных числах параметров для PINN и PIKAN.

В таком случае возникнет выбор устройства, на котором будет проводиться обучение – обычно это или CPU, или GPU. Заметим, что в рамках данной работы алгоритм L-BFGS всегда считался на центральном процессоре, поэтому сравнение актуально для процесса

обучения на фазе алгоритма Adam и запуске модели для предсказаний.

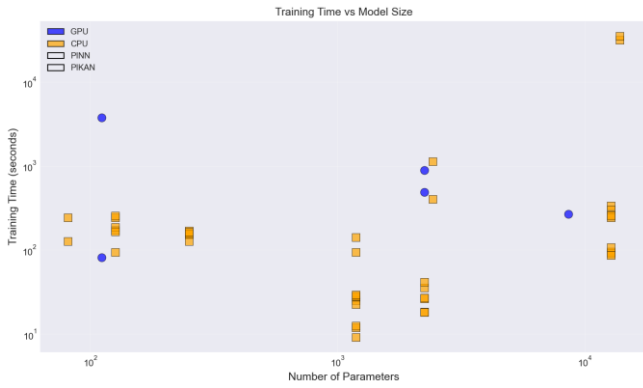


Рис. 5. Сравнение обучения на CPU и GPU

Вычислительные эксперименты проводились под управлением операционной системы Ubuntu. Аппаратная конфигурация включала 6-ядерный процессор AMD Ryzen 5 3600 (базовая частота 3.6 ГГц) с 64 ГБ оперативной памяти DDR4 (3.6 ГГц) и графический ускоритель NVIDIA GeForce GTX 1080 Ti с 11 ГБ видеопамяти.

Теоретическая пиковая производительность используемого GPU для операций одинарной точности (FP32) составляет порядка 11.3 TFLOPS. Производительность процессора Ryzen 5 3600 в аналогичных задачах значительно ниже и находится в диапазоне 0.4–0.6 TFLOPS (в зависимости от использования AVX-инструкций), что демонстрирует разрыв в вычислительной мощности более чем в 20 раз.

Такой контраст критичен для фазы обучения с использованием алгоритма Adam, где высокая степень параллелизма GPU позволяет существенно ускорить расчет градиентов. Однако, как было отмечено ранее, алгоритм L-BFGS выполнялся на CPU. Также отметим сложность вычислений B-сплайнов для PIKAN, что и является основной причиной его отставания от PINN. Наличие новых библиотек, оптимизированных под архитектуру PIKAN, могли бы существенно сократить этот разрыв.

Исходя из графиков, можно сделать следующие выводы:

1. Для PIKAN, GPU существенно замедлил процесс обучения для меди с сопоставимой итоговой сходимостью, но ускорил для диоксида ванадия, при улучшенной точности.
2. Для PINN, CPU всегда был быстрее, чем GPU, в процессе обучения для схожего числа параметров. Для выводов о точности нужны дополнительные запуски скрипта и сбор дополнительных данных.

Заметим, что для PIKAN наблюдается тенденция в улучшении точности в зависимости от числа параметров с переходом в плато на уровне  $10^{-3}$ , поэтому вероятно, что замечание 1 можно интерпретировать как то, что GPU наоборот ухудшил точность PIKAN для меди и не повлиял на неё для диоксида ванадия.

Сравнение PINN и PIKAN по времени обучения при схожем числе параметров можно представить следующим выражением:

$$t(Cu_{KAN}^{CPU}) \leq t(VO2_{PINN}^{GPU}) \leq t(Cu_{PINN}^{GPU}) \leq t(VO2_{KAN}^{CPU}) \quad (12)$$

Также использование GPU дало наилучшую точность для меди и диоксида ванадия для архитектуры KAN в сравнении с запусками PINN и PIKAN на CPU при сопоставимом числе параметров модели.

Необходимо проверить как будет вести себя точность предсказаний для PIKAN с увеличением числа параметров и/или итераций при обучении. Уже после сбора соответствующих данных можно проверить наличие достижения точки оптимума и соответственно сравнить два вычислителя между собой по точности и скорости обучения модели.

### В. Поведение сходимости в течение обучения

Также были получены следующие эмпирические данные:

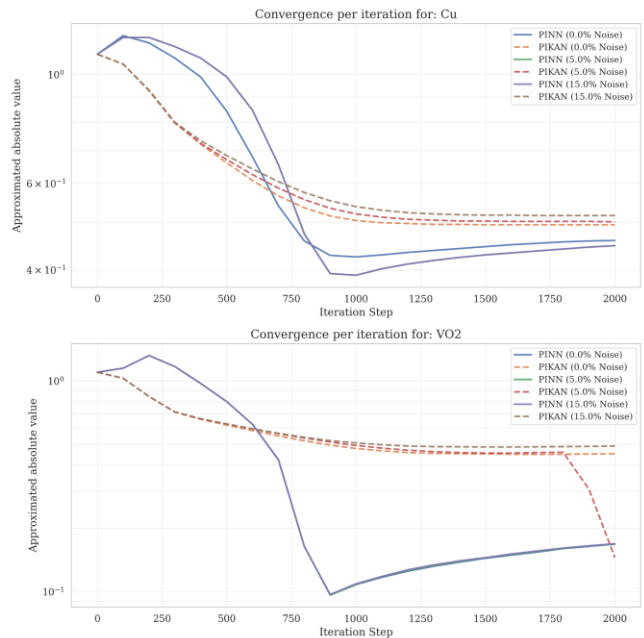


Рис. 6. Изменение предсказания величины  $\alpha$  в течение итераций обучения

На рис. 6 видно, что в большинстве случаев сходимость осуществлялась задолго до начала работы второго алгоритма, а именно в районе 900 итерации. Можно заметить резкое изменение предсказания для PIKAN на диоксиде ванадия с зашумлённостью 5% в районе 1800 итерации – требуется большее число повторений эксперимента для установления причин. Предположим, что дело в удачной конфигурации случайных точек, по которым шло обучение.

В приложении (рис. 13 и 14) можно отследить изменение абсолютного значения, предсказанного моделями, для двух материалов. Ниже на рис. 7 и 8, видно графики разницы в этих значениях между PINN и PIKAN:

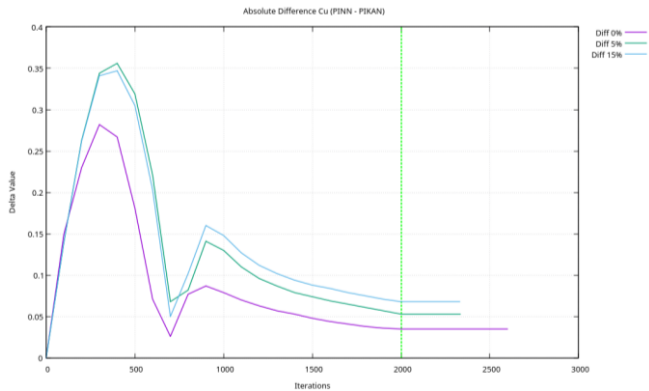


Рис. 7. Дельта ошибки PINN-PIKAN для Cu

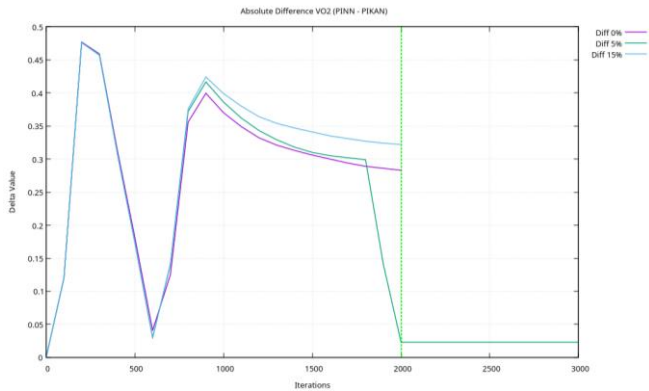


Рис. 8. Дельта ошибки PINN-PIKAN для VO2

По этим графикам становится заметно, что несмотря на меньшее число параметров модели, PIKAN всегда или точнее, или сопоставим с PINN, причём лучше всего это заметно на более зашумлённых данных.

Оба алгоритма склонны к сильному выравниванию предсказываемой величины после L-BFGS, поэтому, в рамках этой работы, число итераций, приходящихся на него, было выбрано значительно меньшим, чем для Adam.

### С. Тепловые карты – медь (Cu)

По итогу обучения, модели решали обратную задачу по восстановлению поведения материала при остывании стержня, а именно восстанавливался коэффициент температуропроводности  $\alpha$  по зашумлённым синтетическим замерам температуры.

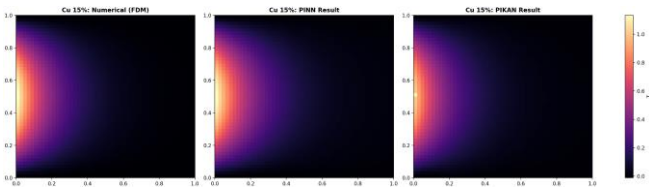


Рис. 9. Предсказание остывания стержня из меди (Cu) с шумом 15%, слева направо: численными методами (FDM), архитектурами PINN и PIKAN

Для упрощения восприятия полученных данных были нарисованы карты распределения тепла и ошибок относительно эталона (FDM).

Все карты есть в разделе Приложение, а также в высоком разрешении в репозитории проекта [19].

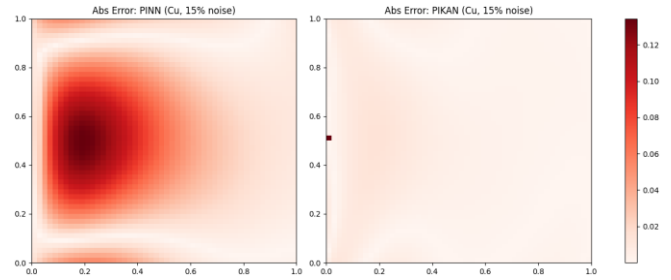


Рис. 10. Карта ошибки для меди (Cu) с шумом 15% в сравнении с численными методами (FDM): слева архитектура PINN, справа - PIKAN

На уровне шума 15% PIKAN продемонстрировал заметно лучшую точность, чем PINN, несмотря на порядок меньшее число параметров. Одновременно с этим, можно обратить внимание на точку ошибки в самом начале временной шкалы для PIKAN, в центре стержня: модель успешно предсказала окрестность пиковой температуры, но ошиблась с абсолютным значением в ней. Этот факт демонстрирует потенциал PIKAN отлавливать фазовые переходы в материалах, ведь архитектурно PINN сглаживает подобные резкие скачки температур, и полностью избавиться от сглаживания не представляется возможным.

### D. Тепловые карты - диоксид ванадия ( $VO_2$ )

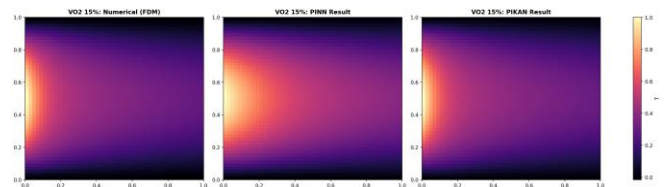


Рис. 11. Предсказание остывания стержня из диоксида ванадия ( $VO_2$ ) с шумом 15%, слева направо: численными методами (FDM), архитектурами PINN и PIKAN

На данной тепловой карте разница в качестве предсказаний становится ещё заметнее. Неспособность PINN не сглаживать показания приводит к образованию аттрактора ошибок – из-за него эта область становится источником ошибочных предсказаний на всём времени эксперимента. Ошибки PIKAN локализованы, причём на карте ошибок, заметны кривые верных решений, похожих на рябь, покрывающих значительную часть графика.

Локальность сплайнового представления в PIKAN играет роль естественного регуляризатора: возмущения от зашумлённых точек влияют лишь на ограниченный набор контрольных точек, не нарушая глобальной физической согласованности.

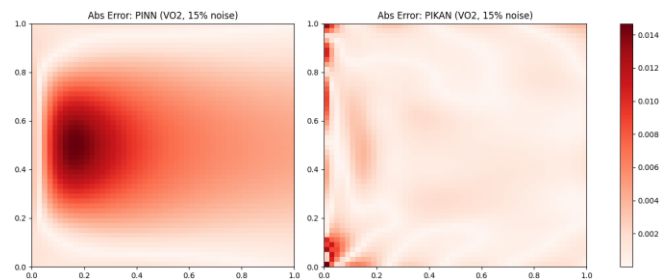


Рис. 12. Карта ошибки для диоксида ванадия ( $VO_2$ ) с шумом 15% в сравнении с численными методами (FDM), слева архитектура PINN, справа PIKAN

## Е. Анализ сходимости

Для меди обе архитектуры показали сопоставимую точность – она представлена в табл. 1:

ТАБЛИЦА I. ОШИБКА ПРЕДСКАЗАНИЯ МОДЕЛЕЙ ДЛЯ МЕДИ

Модель	0%	5%	15%
PINN	$1.27 \times 10^{-4}$	$4.07 \times 10^{-4}$	$1.88 \times 10^{-3}$
PIKAN	$9.48 \times 10^{-5}$	$2.78 \times 10^{-4}$	$1.47 \times 10^{-3}$

Для диоксида ванадия PIKAN был на порядок точнее (табл. 2):

ТАБЛИЦА II. ОШИБКА ПРЕДСКАЗАНИЯ МОДЕЛЕЙ ДЛЯ ДИОКСИДА ВАНАДИЯ

Модель	0%	5%	15%
PINN	$7.24 \times 10^{-3}$	$7.64 \times 10^{-3}$	$1.09 \times 10^{-2}$
PIKAN	$4.23 \times 10^{-4}$	$1.30 \times 10^{-3}$	$3.36 \times 10^{-3}$

Несмотря на повышенные вычислительные затраты на сплайны, применение PIKAN оправдано, когда требуется точность предсказаний поведения материала.

Динамика сходимости функции потерь для обеих архитектур может быть выражена формулой:

$$E_{L_2} = \frac{\|T - T_{pred}\|_2}{\|T\|_2} \quad (13)$$

где  $E_{L_2}$  – ошибка,  $T$  – истинное значение,  $T_{pred}$  – предсказанное значение,  $L_2$  – используемая норма.

Для количественной оценки качества аппроксимации используется относительная норма ошибки  $L_2$ . В отличие от абсолютных метрик, данная норма учитывает масштаб значений искомой функции, что позволяет корректно сопоставлять результаты для материалов с существенно различающимися коэффициентами (Cu и  $VO_2$ ).

PIKAN достигает низких значений потерь быстрее (по числу итераций), хотя каждая итерация может требовать больше вычислительного времени из-за вычисления сплайнов. Тем не менее, общее время до достижения критерия точности у PIKAN оказывается меньше благодаря более быстрой адаптации к высокочастотным компонентам.

## V. ОБСУЖДЕНИЕ

Архитектура PIKAN способна фиксировать резкие градиенты; по числу итераций при обучении сходится быстрее, а также демонстрирует лучшую точность при сопоставимом числе параметров. Помимо этого, В-сплайны обладают свойством интерпретируемости - в отличие от моделей PINN, представленных в виде таблиц чисел. К сожалению, существует недостаток оптимизации библиотек и программного обеспечения при работе с PIKAN, что делает затруднительным использование данной архитектуры в реальных задачах. Однако вышеперечисленные преимущества и потенциал PIKAN для применения в инженерных и исследовательских задачах достаточно вески, чтобы уделить внимание разработке подобных вычислительных алгоритмов, в том

числе в виде исследований на тему применения новых подходов при вычислениях В-сплайнов или разработки новых гибридных архитектур на базе KAN.

Задачи проектирования и диагностики оборудования по показаниям датчиков; Исследования аномальных сред, воспроизведение которых в лабораторных условиях не представляется возможным; Сфера образования, где возможность работы с дорогостоящими сплавами могла бы стать доступнее – качественное, быстрое и точное моделирование сред благодаря сетям Колмогорова-Арнольда может положительно отразиться на всех вышеперечисленных областях жизни общества.

## VI. ЗАКЛЮЧЕНИЕ

Проведённое исследование подтверждает перспективность применения сетей Колмогорова-Арнольда в физически-информированном машинном обучении. Архитектура PIKAN обеспечивает более высокую точность при меньшем числе параметров, эффективно разрешает резкие градиенты и обладает повышенной устойчивостью к шумам измерений. Дальнейшее развитие работы может быть направлено на многомерные обобщения, исследование различных типов сплайнов (вэйвлеты, рациональные функции) и интеграцию с методами reinforcement learning для оптимизации вычислительных траекторий.

Также требуются исследования по выявлению точки оптимума в вопросе числа итераций и параметров при обучении сетей PIKAN под различные типы задач.

Помимо этого, исследователь Ziming Liu, написавший в составе команды библиотеку для работы с данными сетями (PyKAN), утверждает, что интерес может представлять создание новой объединённой архитектуры на базе PINN и PIKAN [18].

## БЛАГОДАРНОСТЬ

Автор выражает благодарность своему научному руководителю Медведеву А.Н. за содействие в изучении алгоритмов глубокого обучения и помощь в постановке физической задачи.

Благодарит Кузьмину Н.Н. за помощь в подготовке документации и рецензировании статьи.

Автор признателен Касимовой В.А. за помощь в техническом оформлении рукописи и приведении текста в соответствие с требованиями ГОСТ.

Отдельную благодарность автор выражает коллективу Мюнхенского Технологического Университета за создание и обновление электронного учебника по машинному обучению для моделирования в физике, а также Ziming Liu – автору репозитория GitHub [18], чей код был использован для работы с архитектурой PIKAN.

## СПИСОК ЛИТЕРАТУРЫ

- [1] Courant R., Friedrichs K., Lewy H. Über die partiellen Differenzgleichungen der mathematischen Physik // Mathematische Annalen. 1928. Vol. 100, No. 1. P. 32-74.
- [2] Raissi M., Perdikaris P., Karniadakis G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations // Journal of Computational Physics. 2019. Vol. 378. P. 686-707.
- [3] Karniadakis G.E. et al. Physics-informed machine learning // Nature Reviews Physics. 2021. Vol. 3, No. 6. P. 422-440.

- [4] Rahaman N. et al. On the spectral bias of neural networks // International Conference on Machine Learning (ICML). PMLR, 2019. P. 5301-5310.
- [5] Liu Z. et al. KAN: Kolmogorov-Arnold Networks // ICLR 2025. URL: <https://openreview.net/forum?id=Ozo7qJ5vZi>.
- [6] Patra S. et al. Physics Informed Kolmogorov-Arnold Neural Networks for Dynamical Analysis via Efficient-KAN and WAV-KAN // Journal of Machine Learning Research. 2025. Vol. 26, No. 233. P. 1-39.
- [7] Wang Y., Xu Y. et al. Scientific machine learning: Kolmogorov-Arnold-Informed Neural Network (KINN) for solving PDEs // arXiv preprint arXiv:2406.11045. 2024.
- [8] TUM Physics-based Simulation Group. Official Website. URL: <https://ge.in.tum.de/>.
- [9] Physics-based Deep Learning Textbook. URL: <https://physicsbaseddeeplearning.org/>
- [10] PIKAN GitHub Repository (Yong Xu et al.). AI and Quantum Computing Group. URL: <https://github.com/AI-and-Quantum-Computing/PIKAN>
- [11] Thermal Conductivity of VO<sub>2</sub> Nanowires at Metal-Insulator Transition Temperature // PMC. 2021. URL: <https://pubmed.ncbi.nlm.nih.gov/articles/PMC8472604/>.
- [12] Enhancement of thermal conductivity across the metal-insulator transition in vanadium dioxide // Applied Physics Letters. 2018. Vol. 113, No. 6. P. 061902. URL: <https://doi.org/10.1063/1.5042089>.
- [13] Kolmogorov A.N. On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables // Doklady Akademii Nauk SSSR. 1956. Vol. 108. P. 179-182.
- [14] Lu L. et al. DeepXDE: A deep learning library for solving differential equations // SIAM Review. 2021. Vol. 63, No. 1. P. 208-228.
- [15] Kiyani E. et al. Optimizing the optimizer for physics-informed neural networks and Kolmogorov-Arnold networks // Computer Methods in Applied Mechanics and Engineering. 2025. Vol. 446. Art. 118308.
- [16] Kingma D.P., Ba J. Adam: A method for stochastic optimization // arXiv preprint arXiv:1412.6980. 2014.
- [17] Wang S., Teng Y., Perdikaris P. Understanding and mitigating gradient flow pathologies in physics-informed neural networks // SIAM Journal on Scientific Computing. 2021. Vol. 43, No. 5. P. A3055-A3081.
- [18] <https://github.com/KindXiaoming/pykan>
- [19] <https://github.com/Comfortropic/PINN-PIKAN-comparison-Cu-VO2-paper/>

## ПРИЛОЖЕНИЕ

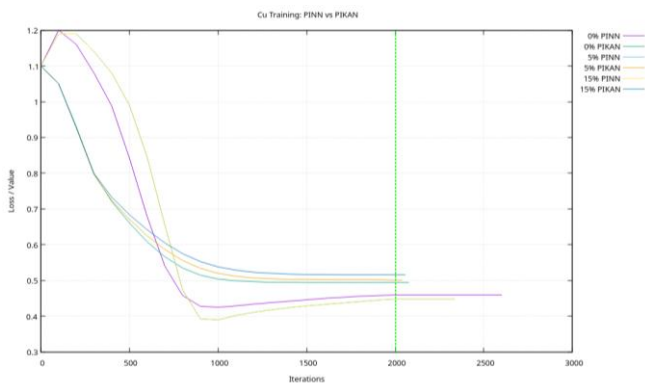


Рис. 1. График изменения абсолютного значения предсказания моделей при обучении для меди (Cu). 2000 итерация - граница работы алгоритмов Adam и L-BFGS.

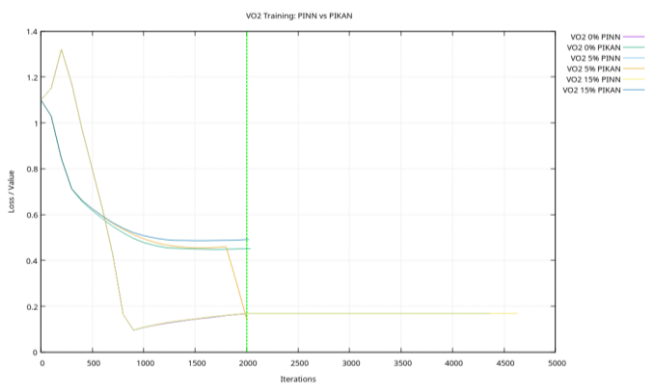


Рис. 2. График изменения абсолютного значения предсказания моделей при обучении для диоксида ванадия (VO<sub>2</sub>). 2000 итерация - граница работы алгоритмов Adam и L-BFGS.

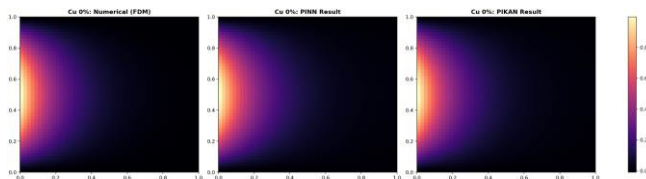


Рис. 3. Предсказание остывания стержня из меди (Cu) без шума, слева направо: численными методами (FDM), архитектурами PINN и PIKAN

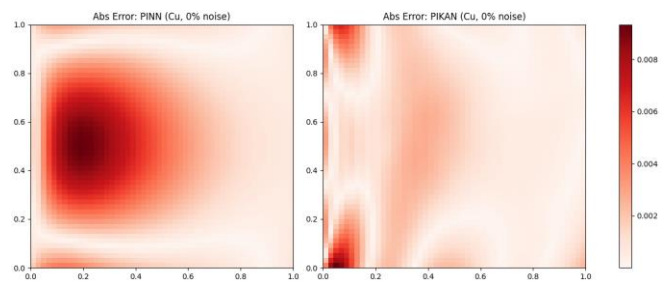


Рис. 4. Карта ошибки для меди (Cu) без шума в сравнении с численными методами (FDM), слева архитектура PINN, справа PIKAN

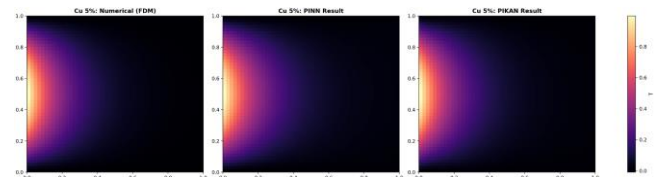


Рис. 5. Предсказание остывания стержня из меди (Cu) с шумом 5%, слева направо: численными методами (FDM), архитектурами PINN и PIKAN

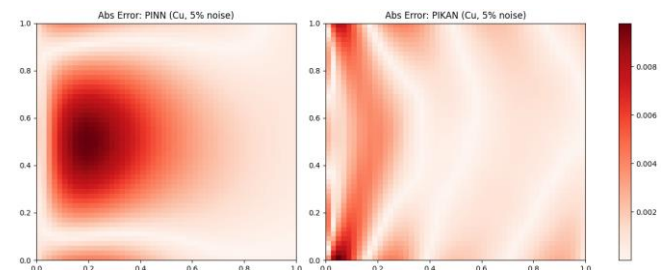


Рис. 6. Карта ошибки для меди (Cu) с шумом 5% в сравнении с численными методами (FDM), слева архитектура PINN, справа PIKAN

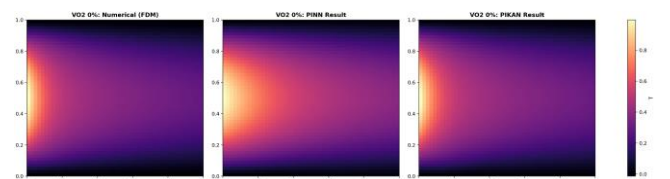


Рис. 7. Предсказание остывания стержня из диоксида ванадия (VO<sub>2</sub>) без шума, слева направо: численными методами (FDM), архитектурами PINN и PIKAN

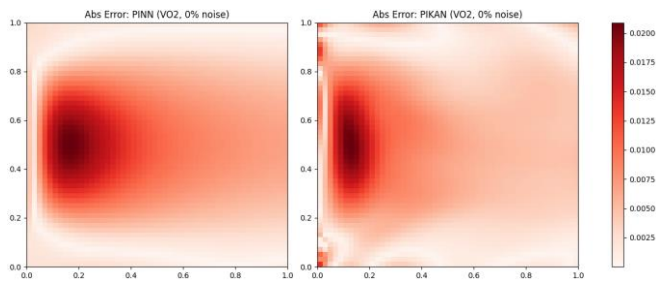


Рис. 8. Карта ошибки для диоксида ванадия (VO2) без шума в сравнении с численными методами (FDM), слева архитектура PINN, справа PIKAN

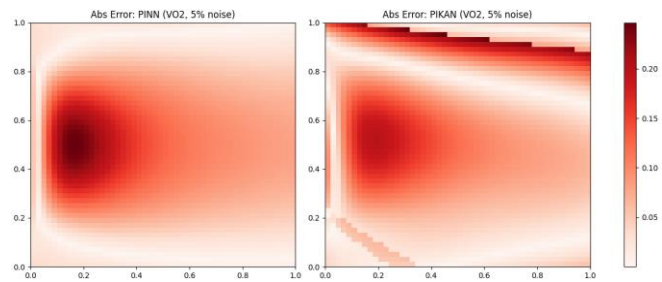


Рис. 10. Карта ошибки для диоксида ванадия (VO2) с шумом 5% в сравнении с численными методами (FDM), слева архитектура PINN, справа PIKAN

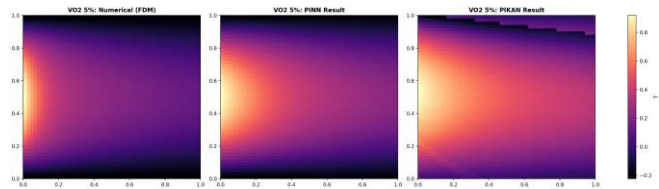


Рис. 9. Предсказание остывания стержня из диоксида ванадия (VO2) с шумом 5%, слева направо: численными методами (FDM), архитектурами PINN и PIKAN