УДК 519.8
DOI: 10.26102/2310-6018/2026.62.11.013

Гибридный подход на основе сетей Колмогорова-Арнольда и генетического программирования для синтеза законов распределения вероятностей для стохастических алгоритмов оптимизации

Е.А. Сопов[1,2]✉, А.Е. Сопов[1], З.Б. Новиков[2]

[1]Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнёва, Красноярск, Российская Федерация
[2]Сибирский федеральный университет, Красноярск, Российская Федерация

Резюме. В работе рассматривается проблема повышения эффективности стохастических алгоритмов глобальной оптимизации, таких как дифференциальная эволюция, за счет адаптивного синтеза законов распределения вероятностей для их управляющих параметров. Существующие методы символьной регрессии для синтеза таких распределений, в частности, генетическое программирование, сталкиваются со сложностями при работе с многомерными данными, в то время как современные нейронные сети Колмогорова-Арнольда, хотя и эффективны для аппроксимации, не обеспечивают точного символьного представления. Для преодоления этих недостатков авторами предложен гибридный метод KAN-GP, в котором сети Колмогорова-Арнольда используются для построения общей структуры регрессионной модели, а генетическое программирование – для точного символьного описания одномерных функций в ее узлах. Эффективность предложенного подхода оценена на классических тестовых задачах символьной регрессии и реальных наборах данных, где он показал более высокую точность по сравнению с каждым из базовых методов. Практическая значимость продемонстрирована на задаче синтеза распределений для алгоритма L-SHADE. Использование разработанного гибридного подхода позволило получить новые законы, применение которых привело к статистически значимому улучшению результатов на множестве эталонных тестовых задач оптимизации IEEE CEC 2017. Предложенный подход может служить основой для конструирования как универсальных, так и специализированных алгоритмов оптимизации, способных автоматически подстраиваться под специфику решаемой оптимизационной задачи.

Ключевые слова: синтез алгоритмов оптимизации, символьная регрессия, генетическое программирование, сети Колмогорова-Арнольда, эволюционные алгоритмы, дифференциальная эволюция.

Финансирование: Исследование выполнено за счет гранта Российского научного фонда № 24-21-00215, https://rscf.ru/project/24-21-00215/.

Для цитирования: Сопов Е.А., Сопов А.Е., Новиков З.Б. Гибридный подход на основе сетей Колмогорова-Арнольда и генетического программирования для синтеза законов распределения вероятностей для стохастических алгоритмов оптимизации. *Моделирование, оптимизация и информационные технологии.* 2026;14(11). URL: https://moitvivt.ru/ru/journal/article?id=2149 DOI: 10.26102/2310-6018/2026.62.11.013

A hybrid approach based on Kolmogorov-Arnold networks and genetic programming for the synthesis of probability distributions in stochastic optimization algorithms

E.A. Sopov[1,2]✉, A.E. Sopov[1], Z.B. Novikov[2]

[1]Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, the Russian Federation

© Сопов Е.А., Сопов А.Е., Новиков З.Б., 2026

[2]*Siberian Federal University, Krasnoyarsk, the Russian Federation*

Abstract. This paper addresses the problem of enhancing the efficiency of stochastic global optimization algorithms, such as differential evolution, through the adaptive synthesis of probability distribution laws for their control parameters. Existing symbolic regression methods for synthesizing such distributions, particularly genetic programming, face challenges when working with high-dimensional data, while modern Kolmogorov-Arnold networks, although effective for approximation, do not provide an exact symbolic representation. To overcome these shortcomings, the authors propose a hybrid KAN-GP method, in which Kolmogorov-Arnold networks are used to construct the overall structure of the regression model, and genetic programming is employed for the precise symbolic description of one-dimensional functions at its nodes. The effectiveness of the proposed approach was evaluated on classical symbolic regression benchmark problems and real-world datasets, where it demonstrated higher accuracy compared to each of the base methods. Its practical significance was demonstrated on the task of synthesizing distributions for the L-SHADE algorithm. The use of the developed hybrid approach enabled the derivation of new laws, the application of which led to a statistically significant improvement in results on a set of benchmark optimization problems from IEEE CEC 2017. The proposed approach can serve as a foundation for constructing both universal and specialized optimization algorithms capable of automatically adapting to the specifics of the optimization problem being solved.

Keywords: optimization algorithm synthesis, symbolic regression, genetic programming, Kolmogorov-Arnold networks, evolutionary algorithms, differential evolution.

Funding: The study was supported by grant No. 24-21-00215 from the Russian Science Foundation, https://rscf.ru/project/24-21-00215/.

For citation: Sopov E.A., Sopov A.E., Novikov Z.B. A hybrid approach based on Kolmogorov-Arnold networks and genetic programming for the synthesis of probability distributions in stochastic optimization algorithms. *Modeling, Optimization and Information Technology*. 2026;14(11). (In Russ.). URL: https://moitvivt.ru/ru/journal/article?id=2149 DOI: 10.26102/2310-6018/2026.62.11.013

Введение

Многие прикладные задачи глобальной оптимизации заданы алгоритмически или являются настолько сложными, что рассматриваются как модель «черного ящика». Для решения подобных задач предложены различные алгоритмы нулевого порядка, среди которых стоит выделить стохастические популяционные алгоритмы, включая эволюционные алгоритмы [1]. Для этого класса алгоритмов характерно применение операций поиска, которые используют те или иные законы распределения вероятностей и, как показано в ряде исследований, выбор и настройка параметров закона распределения существенно влияет на эффективность решения задачи [2].

Для задач с вещественными переменными одним из наиболее эффективных алгоритмов оптимизации является адаптивная дифференциальная эволюция на основе истории успеха (SHADE) [3]. Алгоритмы, основанные на SHADE, генерируют управляющие параметры для поисковых операций с помощью фиксированных законов распределения вероятностей, которые могут оказаться неэффективными для конкретных экземпляров или классов задач. Таким образом, актуальной является задача синтеза произвольных распределений, способных повысить эффективность работы подобных алгоритмов.

В данном исследовании под законом распределения вероятностей мы будем подразумевать алгоритм, назначающий вероятности для выбора значений управляемого параметра. Для вещественных параметров такой алгоритм представляет собой функцию, которую можно получить в ходе решения задачи регрессии методами машинного обучения. Для задачи синтеза законов распределения вероятностей более перспективным является построение объяснимых моделей в форме символьных

выражений (формул). Такие выражения можно проанализировать и, при необходимости, подвергнуть математическим преобразованиям. Эта задача известна как обнаружение символьных формул или символьная регрессия [4]. Одним из самых популярных подходов для решения этой задачи является генетическое программирование (Genetic Programming, GP) с представлением решений в виде дерева[1]. К недостаткам этого метода относится его низкая производительность на данных высокой размерности и со сложными нелинейными зависимостями.

Недавно был предложен новый тип нейронных сетей – сети Колмогорова-Арнольда (Kolmogorov-Arnold Network, KAN), которые предлагают эффективный подход для решения задач машинного обучения, обеспечивая не только высокую точность, но и внутреннюю интерпретируемость. Эта интерпретируемость достигается за счет упрощения (pruning) обученной сети KAN и нахождения символьных представлений для активационных функций в узлах сети. Обычно эти узлы представлены простыми сплайнами. Однако вместо получения точного символьного представления стандартный подход в KAN выбирает наиболее похожую функцию из заранее заданного набора элементарных функций (например, x, x^2, sin, cos, log, exp) [5], что порождает удобочитаемые, но не точные в символьном отношении выражения.

В данном исследовании мы предлагаем гибридный подход KAN-GP для повышения символьной точности сетей KAN. Поскольку любой узел KAN представляет собой функцию одной переменной, мы можем использовать GP для обнаружения его точного символьного представления. Предлагаемый подход сочетает в себе способность KAN к обобщению и высокую скорость обучения для работы со сложными многомерными данными с символьной точностью генетического программирования, которая в рамках предложенного подхода достигается за счет решения одномерных задач.

Для оценки эффективности мы сравнили производительность гибрида KAN-GP с традиционными методами KAN и генетического программирования на тестовых задачах символьной регрессии и некоторых реальные наборах данных. Далее мы апробировали подход на задаче синтеза законов распределения вероятности для алгоритма SHADE, эффективность которого оценили на множестве эталонных тестовых задач оптимизации IEEE CEC 2017.

Материалы и методы

Законы распределения вероятностей в самоадаптивной дифференциальной эволюции. Алгоритм дифференциальной эволюции (ДЭ) [6], предложенный Шторном и Прайсом в 1995, остается в фокусе внимания научного сообщества благодаря своей эффективности и относительной простоте реализации [7]. Различные модификации ДЭ демонстрируют стабильно высокие результаты при решении сложных задач глобальной оптимизации «черного ящика». Это подтверждается, в том числе, статистикой международных конкурсов по оптимизации, в частности, IEEE Congress on Evolutionary Computation (CEC), где большинство представленных алгоритмов-лидеров основаны на ДЭ [8].

В настоящее время, основное направление исследований ДЭ связано с разработкой подходов, способных автоматически настраивать два ключевых параметра: коэффициент масштабирования F (параметр операции мутации) и вероятности рекомбинации CR, значения которых существенно влияют на эффективность алгоритма. Также важным параметром является размер популяции, однако в последние годы этот

[1] Poli R., Langdon W.B., McPhee N.F. *A field guide to genetic programming*. Lulu Enterprises Uk Ltd; 2008. 233 p. URL: http://www.gp-field-guide.org.uk

вопрос практически решен за счет использования методов уменьшения размера популяции [9].

Одними из первых подходов, предложивших адаптивную схему настройки параметров F и CR, являются алгоритмы SaDE [10] и jDE [11], которые использовали случайный выбор параметров на основе нормального закона распределения вероятностей. Исследования показали, что даже простая рандомизация параметров может повысить эффективность по сравнению с выбором статических значений. В алгоритме JADE [12] было предложено запоминать успешные значения случайно сгенерированных параметров. Наконец, на основе объединения ранее предложенных подходов были разработаны адаптивный алгоритм на основе истории успеха SHADE (Success-History based Adaptive Differential Evolution) [13] и его модификация L-SHADE [9] с процедурой уменьшения размера популяции, которые породили целое направление развития алгоритмов, использующих идею анализа истории успеха. SHADE и многие его производные алгоритмы используют следующую схему определения параметров F и CR:

$$\begin{cases} F = RandomCauchy(M_F; 0{,}1) \\ CR = RandomNormal(M_{CR}; 0{,}1) \end{cases}, \quad (1)$$

где M_F и M_{CR} – коэффициент сдвига распределения Коши и математическое ожидание нормального распределения, выбранные из архива успешных значений.

Выбор данных распределений обосновывают тем, что нормальный закон более избирательный, чаще генерирует значения, близкие к математическому ожиданию, а распределение Коши имеет более тяжелые хвосты, что полезно для операции мутации. В то же время было показано, что выбор других распределений может повысить эффективность алгоритма [14]. Очевидно, что помимо традиционных распределений из математической статистики можно синтезировать и другие алгоритмы назначений вероятностей. Поскольку параметры F и CR вещественные в интервале [0,1], подобный алгоритм может быть представлен как функциональная зависимость по аналогии с представлением плотности распределения вероятностей. Данная задача синтеза закона распределения может быть решена как задача регрессии. В отличие от традиционной регрессии, основанной на минимизации среднеквадратической ошибки, в предложенном подходе оценкой качества будет оценка эффективности алгоритма оптимизации, который использует закон распределения, представленный уравнением регрессии.

Одним из первых методов индуктивного поиска произвольных аналитических выражений стал метод группового учета аргументов [15], основанный на использовании полиномов Колмогорова-Габора. Алгоритм начинает с простых одномерных структур и последовательно усложняет их, отбирая наиболее точные полиномиальные представления. Несмотря на высокую прогностическую точность, принципиальным ограничением метода группового учета аргументов остается невозможность выхода за рамки полиномиальной структуры.

Современное понятие символьной регрессии неразрывно связано с профессором Джоном Коза, разработавшего метод генетического программирования (ГП) [16, 17]. Данный подход, реализующий автоматизированный синтез компьютерных программ, оказался весьма эффективным и для решения задач символьной регрессии. С момента появления канонического ГП было предложено множество его модификаций [18, 19]. Сравнительное исследование [20], охватившее более 20 популярных алгоритмов на 250 наборах данных, показало, что методы на основе ГП не только превосходят альтернативные подходы, но и демонстрируют конкурентоспособность с современными методами глубокого обучения.

В основе генетического программирования лежит использование эволюционных алгоритмов, осуществляющих поиск оптимального решения в пространстве возможных

выражений, формируемых путем суперпозиции функциональных и терминальных элементов. При решении задач символьной регрессии функциональное множество включает математические операции и функции, а терминальное множество состоит из независимых переменных и констант. Несмотря на существование различных способов представления выражений, наиболее распространенным остается древовидное представление.

Главной проблемой древовидной структуры является явление «разбухание» деревьев (tree bloating) [21], затрудняющее нахождение точных символьных решений. Для иллюстрации: даже компактное представление простой функции $f(x) = x^1 + x^2 + \cdots + x_{10}$ требует дерева глубиной 5 с 8 функциональными и 10 терминальными узлами. По мере усложнения решаемой задачи размер дерева может расти экспоненциально, что существенно затрудняет как вычислительные процедуры, так и интерпретацию полученных решений.

Другой фундаментальной проблемой является специфика ландшафта пространства поиска. В отличие от традиционных методов регрессии, где малые изменения настраиваемых параметров (весов) приводят к пропорциональным изменениям точности модели, в ГП минимальная модификация генотипа (например, замена одной операции) может вызвать существенное изменение в итоговой функции и ее пригодности. Например, замена операции сложения на деление в выражении $f^1(x) = 10 \cdot cos(x) + (x - 2)$ приводит к функции $f_2(x) = (10 \cdot cos(x))/(x - 2)$. Эти функции являются ближайшими соседями в пространстве поиска, т. к. мы сделали минимально возможное изменение, но графики функции отличаются сильно, также новая функция теперь может порождать некорректные операции (деление на ноль). Данное свойство пространства поиска ГП, характеризующееся наличием «рваных» участков и резких перепадов, принципиально отличает его от гладких пространств поиска в традиционной параметрической оптимизации.

Сети Колмогорова-Арнольда (KAN) представляют собой новейшую архитектуру нейронных сетей, основанную на теореме Колмогорова-Арнольда [5]. Согласно данной теореме, любую многомерную непрерывную функцию можно представить в виде конечной композиции непрерывных функций одной переменной. Это фундаментальное положение кардинально отличает KAN от традиционных многослойных перцептронов (MLP): вместо обучения весовых коэффициентов при фиксированных функциях активации, KAN используют обучаемые активационные функции, расположенные на ребрах сети.

Ключевой особенностью KAN является параметризация активационных функций в виде сплайнов, которые впоследствии могут быть аппроксимированы компактными символьными выражениями (такими как x^2, $sin(x)$). Данное свойство выгодно отличает KAN от «черного ящика» традиционных MLP. Кроме того, KAN демонстрируют высокую точность при использовании градиентных методов обучения, достигая сравнимых результатов с существенно меньшим количеством обучаемых параметров [22, 23].

Решение задачи символьной регрессии с использованием KAN включает несколько этапов. Первоначально сеть обучается на предоставленном наборе данных, после чего подвергается процедуре упрощения (pruning), в ходе которой удаляются избыточные связи, оказывающие минимальное влияние на точность. Благодаря изначально простой структуре KAN, полученная после упрощения сеть обычно содержит небольшое количество узлов и соединений.

Обычно сплайновые аппроксимации в узлах сети отличаются гладкостью и часто соответствуют элементарным функциям – монотонным линейным или квадратичным

зависимостям, выпуклым функциям (например, параболе) или периодическим функциям (синусу, косинусу). Для построения символьного представления обнаруженной зависимости необходимо определить набор примитивных одномерных функций с известными аналитическими представлениями. Для каждого сплайна выбирается наиболее близкая символьная функция, после чего полное символьное выражение строится путем композиции этих функций в соответствии с топологией упрощенной сети.

Несмотря на высокую эффективность при работе со сложными многомерными задачами регрессии, получаемые символьные выражения имеют ограниченную точность. Это связано с тем, что обученные сплайны заменяются простыми элементарными функциями, а не истинными символьными выражениями. К тому же сами сплайны строятся с использованием малого количества опорных точек (3–5), что неизбежно приводит к потере общей точности аппроксимации.

Методы на основе ГП способны находить символьно точные выражения, однако наилучшие результаты достигаются на задачах малой размерности. При работе с многомерными задачами ГП вынуждено формировать большие деревья для включения всех значимых переменных, что существенно усложняет как эволюционный поиск, так и вычислительные процессы. Более того, учитывая, что локальные изменения в структуре дерева могут нарушать накопленную информацию, эффективность ГП падает по мере роста сложности задачи символьной регрессии.

В данном исследовании предлагается гибридный подход, объединяющий преимущества обоих методов. Мы используем сети KAN в качестве эффективных и универсальных регрессоров для обработки многомерных данных, тогда как генетическое программирование применяется как инструмент символьной точности для аппроксимации функций одной переменной в узлах сетей KAN.

Предложенный подход. Предлагаемый гибридный подход KAN-GP может быть представлен в виде следующего алгоритма:

Вход: набор данных, гиперпараметры KAN, настройки ГП.

Выход: символьное выражение.

Шаг 1. Обучить сеть KAN на обучающем наборе данных.

Шаг 2. Упростить структуру KAN и выполнить дополнительное обучение для тонкой донастройки.

Шаг 3. Для каждого узла в упрощенной сети KAN:

3.1. Извлечь активационную функцию (сплайн).

3.2. Сгенерировать данные для обучения символьной регрессии.

3.3. Применить ГП для обнаружения точного символьного представления.

Шаг 4. Построить полное символьное выражение на основе топологии KAN.

На начальном этапе для решения задачи регрессии необходимо выполнить стандартные подготовительные этапы: предварительную обработку данных, подбор гиперпараметров KAN и ГП для достижения удовлетворительной точности, измеряемой среднеквадратической ошибкой (MSE) или другими релевантными метриками.

После обучения сети KAN выполняется процедура упрощения структуры для получения компактной сети без значительной потери точности. Точность упрощенной сети часто может быть улучшена путем дополнительной тонкой настройки. Пороговые значения упрощения сами являются гиперпараметрами метода: малые значения могут сохранять узлы, аппроксимирующие шум в данных, что приводит к громоздким и негладким итоговым выражениям, тогда как высокие значения способствуют лучшей обобщающей способности, но могут снижать точность сети.

На следующем этапе анализируются активационные функции в упрощенной сети. Непрерывный характер этих функций позволяет генерировать обучающую выборку для

ГП с равномерным шагом для точной аппроксимации, даже когда исходные данные распределены неравномерно. Поскольку решается задача одномерной регрессии, мы можем использовать достаточно большой объем выборки без существенных вычислительных затрат.

Как демонстрируют графики активационных функций KAN, эти функции обычно отличаются гладкостью. Данное свойство может быть использовано для управления функциональным множеством алгоритма ГП: увеличивается вероятность выбора операций, генерирующих гладкие функции (сложение, умножение, sin, cos), и снижается вероятность операций, способных приводить к негладкому поведению (деление, модуль).

Для сравнительного анализа различных подходов используются следующие метрики точности, основанные на среднеквадратической ошибке (MSE) обнаруженных символьных выражений: MSE_KAN – точность выражений, извлеченных из упрощенных сетей KAN стандартным методом, MSE_GP – точность выражений, обнаруженных стандартным генетическим программированием, MSE_KAN-GP – точность выражений, обнаруженных предлагаемым гибридным подходом.

Для тестовых задач с известными эталонными выражениями дополнительно оценивается, было ли точное восстановление исходного выражения или, как минимум, корректной функциональной структуры.

Задача синтеза законов распределения вероятностей решается в данной работе для алгоритма L-SHADE, однако подход применим и к любым другим алгоритмам стохастической оптимизации. Общая схема подхода представлена ниже:

Вход: алгоритм оптимизации, задача (набор задач) оптимизации, мера качества решения задачи оптимизации, алгоритм KAN-GP.

Выход: новый закон распределения для алгоритма оптимизации.

Шаг 1. Инициализировать KAN-GP.

Шаг 2. На каждом шаге обучения KAN:

2.1. Использовать полученную модель регрессии KAN для разыгрывания вероятностей при выборе параметров в алгоритме оптимизации.

2.2. Решить задачу оптимизации и оценить эффективность алгоритма оптимизации.

2.3. Использовать оценку эффективности как значение функции потерь в KAN.

Шаг 3. Извлечь итоговое символьное выражение, используя KAN-GP.

На шаге 2.2 происходит оценка эффективности алгоритма оптимизации, использующего синтезированный закон распределения вероятностей. В случае, если закон синтезируется для одной задачи, оценкой эффективности является точность алгоритма, выраженная значением целевой функции лучшего найденного решения. Если закон синтезируется для класса (множества) задач, тогда оценкой эффективности является среднее по оценкам эффективности каждой из задач множества. Для вычислительно сложных задач можно использовать не весь бюджет оптимизации, а оценивать скорость сходимости и прерывать оптимизацию до исчерпания бюджета, но периодически уточнять оценки, используя полный бюджет.

В общем случае в метод ГП можно включить автоматически определяемые функции (Automatically Defined Function, ADF), которые являются элементами терминального множества и содержат готовые функциональные конструкции. В частности, во множество ADF можно включить известные законы распределения.

Реализация подходов, настройки численных экспериментов. Для оценки эффективности предложенного подхода и сравнения его со стандартными методами символьного обнаружения KAN и символьной регрессии ГП все методы были

протестированы на популярных наборах тестовых задач символьной регрессии. Набор данных включает 12 задач из бенчмарка Nguyen [24] и 22 задачи из бенчмарка Livermore [25]. Кроме того, алгоритмы были оценены на пяти реальных наборах данных регрессии из репозитория UCI: «Concrete Compressive Strength», «Combined Cycle Power Plant», «Airfoil Self-Noise», «Yacht Hydrodynamics» и «Synchronous Machine dataset»[2].

Задача синтеза законов распределений вероятностей решалась для алгоритма L-SHADE для набора эталонных тестовых задач глобальной оптимизации «черного ящика» IEEE CEC 2017, который содержит 30 задач[3]. Исследования проводились для размерностей 10 и 30 с бюджетом вычислений 10000*D.

Для реализации KAN использовалась библиотека PyKAN со следующими параметрами:

– Архитектура KAN: $[n, 2n, 1]$ для $n \leq 4$ переменных и $[n, 3n, 2n, 1]$ для $n > 4$ переменных;

– Конфигурация сплайнов: $grid = 5$ и $k = 3$;

– Оптимизация: оптимизатор LBFGS с коэффициентом L_1-регуляризации 0,001;

– Обучение: максимальное количество эпох – 100;

– Упрощение: стандартные пороги PyKAN;

– Параметры генетического программирования.

– Стандартное ГП на основе деревьев реализовано на языке C++ с поддержкой multithreading. Использовались следующие параметры:

– Размер популяции: $40n$;

– Метод инициализации: полный метод с максимальной глубиной 5;

– Метод селекции: турнирная селекция с размером турнира 2;

– Кроссинговер: стандартный одноточечный кроссинговер;

– Мутация: одноузловая мутация с вероятностью 0,01;

– Функциональное множество: {(+;0,24), (−;0,24), (×;0,20), (÷;0,08), (sin;0,08), (cos;0,08), (pow;0,02), (log;0,02), (sqrt;0,02)};

– Терминальное множество: {(variable;0,4), (const;0,6)};

– Функция приспособленности для тестовых задач: на основе MSE со штрафами за сложность и использование защищенных операторов (2), где $\alpha = 0{,}01$ и $\beta = 0{,}1$;

– Функция приспособленности для задачи синтеза распределения: аналогична (2), за исключением оценки регрессии – MSE заменяется на значение целевой функции лучшего найденного решения в оптимизации или среднее по лучшим решениям на множестве задач (рассматривается задача минимизации).

$$fitness(tree) = \frac{1}{1+MSE(tree)} \cdot \sqrt{e^{-\alpha \cdot size(tree)} \cdot e^{-\frac{\alpha}{2} \cdot depth(tree)}} \cdot e^{-\beta \cdot PCount(tree)} \quad (2)$$

Результаты

Результаты экспериментов были получены на основе 40 независимых запусков каждого алгоритма. Для всех тестовых задач вычислялось среднее значение по оценкам среднеквадратической ошибки (СКО). Для бенчмарков Nguyen и Livermore дополнительно указывается процент запусков, в которых была корректно идентифицирована функциональная структура, и процент точного восстановления исходного символьного выражения. Для реальных задач регрессии дополнительно

[2] *UCI Machine Learning Repository.* URL: https://archive.ics.uci.edu/ (дата обращения: 01.11.2025).

[3] Awad N.H., Ali M.Z., Liang J.J., Qu B.Y., Suganthan P.N. *Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective real-parameter numerical optimization. Technical Report.* Singapore: Nanyang Technological University; 2016.

приводится медианное количество узлов и глубина лучших решений, найденных во всех запусках. Результаты представлены в Таблицах 1 и 2.

Таблица 1 – Результаты численных экспериментов на тестовых задачах регрессии
Table 1 – The experimental results for benchmark regression problems

Задача	KAN	GP	KAN-GP
	СКО (процент корректных функциональных структур, процент точных выражений)		
Nguyen-1, Nguyen-2, Nguyen-3, Nguyen-4, Nguyen-9, Livermore-1, Livermore-5, Livermore-6, Livermore-14, Livermore-19, Livermore-20, Livermore-22	0,0 (100; 100)	0,0 (100; 100)	0,0 (100; 100)
Nguyen-5	0,5437 (100; 85)	0,0 (100; 100)	0,4500 (100; 90)
Nguyen-6	0,1675 (92,5; 90)	0,0 (100; 100)	0,1183 (100; 92,5)
Nguyen-7	0,0152 (32,5; 15)	0,0169 (22,5; 22,5)	0,0052 (70; 52,5)
Nguyen-8	0,0539 (72,5; 52,5)	0,0 (100; 100)	0,0 (100; 100)
Nguyen-10	0,1054 (82,5; 70)	0,1938 (70; 70)	0,0 (100; 100)
Nguyen-11	0,5156 (0; 0)	0,0 (100; 100)	0,3017 (0; 0)
Nguyen-12	0,0844 (82,5; 57,5)	0,1101 (42,5; 42,5)	0,0337(100; 87,5)
Livermore-2	0,0062 (85; 47,5)	0,0057 (52,5; 50)	0,0 (100; 100)
Livermore-3	0,0037 (67,5; 42,5)	0,0086 (47,5; 47,5)	0,0032 (100; 72,5)
Livermore-4	1,4457 (72,5; 70)	3,9882 (27,5; 27,5)	0,0 (100; 100)
Livermore-7	0,0035 (0; 0)	0,0113 (0; 0)	0,0034 (0; 0)
Livermore-8	0,0061 (0; 0)	0,0172 (0; 0)	0,0038 (0; 0)
Livermore-9	0,0 (100; 100)	2,2274 (52,5; 22,5)	0,0 (100; 100)
Livermore-10	0,0448 (65; 42,5)	0,0 (100; 100)	0,0 (100; 100)
Livermore-11	0,4178 (85; 65)	0,0 (100; 100)	0,0 (100; 100)
Livermore-12	1833,30 (87,5; 55)	0,0 (100; 100)	347,02 (100; 62,5)
Livermore-13	0,0057 (22,5; 0)	0,0 (100; 100)	0,0054 (22,5; 0)
Livermore-15	0,0062 (20; 0)	0,0 (100; 100)	0,0059 (20; 0)
Livermore-16	0,0053 (35; 0)	0,0 (100; 100)	0,0053 (35; 0)
Livermore-17	0,0069 (70; 52,5)	0,0 (100; 100)	0,0 (100; 100)
Livermore-18	0,0059 (65; 55)	0,0 (100; 100)	0,0 (100; 100)
Livermore-21	0,0 (100; 100)	0,8421 (22,5; 0)	0,0 (100; 100)

Таблица 2 – Результаты численных экспериментов на практических задачах регрессии
Table 2 – The experimental results for real-world regression problems

Набор данных	KAN	GP	KAN−GP
СКО (число узлов, глубина дерева)	201,42 (−; −)	248,11 (53; 7)	200,98 (13; 5)
Concrete Compressive Strength ($n = 8$)	157,84 (−; −)	207,13 (14; 5)	139,99 (20; 6)
Combined Cycle Power Plant ($n = 4$)	38,92 (−; −)	42,56 (52; 7)	37,8 (74; 8)
Airfoil Self−Noise ($n = 5$)	281,76 (−; −)	328,01 (71; 8)	269,33 (92; 8)
Yacht Hydrodynamics ($n = 6$)	0,0027 (−; −)	0,0028 (37; 7)	0,0027 (18; 6)
Synchronous Machine ($n = 4$)	201,42 (−; −)	248,11 (53; 7)	200,98 (13; 5)

Как показывают результаты, многие тестовые задачи одной и нескольких переменных не представляют особой сложности для любого из рассматриваемых методов, и все алгоритмы успешно находят точные символьные выражения. Эти зашумленные тестовые задачи в основном предназначены для оценки способности

алгоритмов символьной регрессии обнаруживать различные типы нелинейных зависимостей, включая полиномиальные, степенные и периодические функции.

Во многих задачах сети KAN демонстрируют способность определять правильную функциональную структуру, но не могут точно определить коэффициенты. В то же время ГП либо находит точное выражение с правильной структурой, либо создает численно точную аппроксимацию, которая является функционально эквивалентной, но символьно отличной от исходной.

На тестовых задачах предложенный подход в целом превосходит как стандартные методы KAN, так и ГП. Однако в редких случаях его эффективность ограничена из-за недостатков начального представления KAN. Например, задача Nguyen-11 ($f(x,y) = x^y$) легко решается ГП, поскольку степенная функция включена в его функциональное множество, а KAN пытается аппроксимировать эту функцию через одномерную декомпозицию, что требует введения более высокой нелинейности посредством углубления архитектуры. В результате обученные сплайны в узлах KAN не могут точно представить полное выражение через простые функции.

Результаты решения задачи проектирования законов распределения показаны в Таблице 3 (указаны среднее значение и стандартное отклонение по серии запусков, значения в ячейках алгоритма с лучшими результатами выделены жирным).

Таблица 3 – Результаты оценки эффективности алгоритма оптимизации с синтезированным законом распределения вероятностей методом KAN-GP
Table 3 – Performance evaluation results of the optimization algorithm with a probability distribution synthesized by the KAN-GP method

Задача	**L-SHADE**		**L-SHADE с новым законом распределения**	
	D = 10	D = 30	D = 10	D = 30
F1	0,00/0,00	0,00/0,00	0,00/0,00	0,00/0,00
F2	0,00/0,00	0,00/0,00	0,00/0,00	0,00/0,00
F3	0,00/0,00	0,00/0,00	0,00/0,00	0,00/0,00
F4	0,00/0,00	0,00/0,00	0,00/0,00	0,00/0,00
F5	1,35/0,71	3,69/2,48	1,10/0,70	3,38/2,64
F6	0,00/0,00	0,00/0,00	0,00/0,00	0,00/0,00
F7	11,00/0,35	33,80/0,82	11,33/0,28	37,15/0,82
F8	1,04/0,74	3,59/1,62	0,87/0,61	3,77/1,17
F9	0,00/0,00	0,00/0,00	0,00/0,00	0,00/0,00
F10	10,50/24,00	1440,00/318,00	10,48/20,09	1520,00/329,64
F11	0,02/0,14	3,79/2,49	0,02/0,11	3,16/1,78
F12	114,00/65,70	554,00/253,00	115,42/63,37	450,51/276,41
F13	3,37/2,42	15,20/4,71	2,74/2,39	13,82/5,08
F14	0,05/0,20	22,80/1,78	0,05/0,18	20,21/1,27
F15	0,41/0,27	5,37/2,80	0,36/0,25	5,49/2,46
F16	0,64/0,23	37,70/55,40	0,59/0,20	29,98/54,18
F17	1,54/4,44	31,70/8,68	1,59/3,85	32,09/6,50
F18	1,97/5,44	23,40/2,00	1,75/4,35	17,79/1,70
F19	0,08/0,21	9,45/2,01	0,07/0,19	8,54/1,87
F20	0,18/0,17	81,60/58,60	0,16/0,16	68,21/54,81
F21	154,00/49,30	209,00/4,19	136,67/40,90	178,24/4,40
F22	100,00/0,08	100,00/0,00	99,89/0,07	81,79/0,00
F23	302,00/1,50	357,00/4,07	242,45/1,40	363,26/3,17
F24	292,00/82,90	429,00/2,38	276,93/74,46	362,91/2,28
F25	426,00/22,40	378,00/0,01	423,48/18,36	311,94/0,01

Таблица 3 (продолжение)
Table 3 (continued)

F26	300,00/0,00	943,00/35,60	278,25/0,00	968,42/34,15
F27	388,00/3,13	497,00/14,10	312,56/2,69	483,83/15,34
F28	363,00/82,90	334,00/51,20	329,31/81,56	261,37/39,93
F29	230,00/2,24	395,00/40,60	238,76/2,05	431,33/43,91
F30	230,00/32,50	401,00/90,60	191,81/28,36	308,87/93,05

Как мы видим из таблицы, алгоритм с синтезированными законами распределения, отличными от стандартных нормального закона и закона Коши в L-SHADE дают в среднем лучшие результаты на наборе эталонных задач оптимизации. Для размерности 10 новый алгоритм превзошел базовый 19 раз и 6 раз получил аналогичный результат, на 5 задачах базовый оказался лучше. Для размерности 30 новый алгоритм превзошел базовый 16 раз, аналогичный результат – 6 раз, проиграл – 8 раз. В некоторых случаях, новый подход не улучшил среднее по запускам, но уменьшил стандартное отклонение, т. е. обеспечил лучшую стабильность, повторяемость результатов.

В бенчмарке CEC17 сложность задач растет от первой к последней. Как мы видим из Таблицы 3, алгоритм с синтезированным законом распределения превосходит базовый на наиболее сложных задачах. Стоит отметить, что при использовании предложенного подхода KAN-GP мы синтезировали закон, обеспечивающий высокую эффективность оптимизации в среднем на наборе. Поскольку для конкретной задачи типа «черный ящик» мы не может заранее определить свойства задачи и выбрать закон распределения под нее, эффективность в среднем гарантирует надежность алгоритма при появлении произвольных задач. Однако для типовых или повторяющихся задач, мы можем применить KAN-GP для создания узкоспециализированного алгоритма оптимизации, обеспечивающего максимум эффективности для данной задачи и не гарантирующего высокую эффективность на других задачах.

Обсуждение

Полученные результаты демонстрируют, что гибридный подход KAN-GP способен находить точные символьные выражения для разнообразных нелинейных и многомерных наборов данных. Однако его эффективность критически зависит от качества обучения исходной сети KAN, что сохраняет актуальность задачи совершенствования базового алгоритма KAN.

При решении прикладных задач регрессии метод KAN демонстрирует превосходство над ГП в смысле среднеквадратической ошибки полученных выражений. Снижение эффективности ГП связано с необходимостью построения больших и глубоких деревьев, включающих все переменные в листовых узлах. В отличие от этого, KAN способен строить гладкие аппроксимации, не страдая от «проклятия размерности».

На тестовых задачах регрессии гибридный подход KAN-GP превосходит оба базовых алгоритма по всем рассматриваемым задачам. Символьные выражения, получаемые с помощью KAN-GP, обычно сложнее выражений стандартного ГП, поскольку итоговое решение комбинирует множество функциональных выражений из узлов KAN. В то же время, выражения внутри отдельных узлов KAN остаются относительно простыми, что способствует лучшей интерпретируемости вклада каждой переменной в общее решение. Более того, эти узловые выражения значительно проще обнаружить, поскольку они включают только одну переменную. Для реальных задач мы

не можем однозначно оценить корректность полученных выражений, однако считаем предпочтительными более точные выражения со сравнимой сложностью.

Проведенные эксперименты в задачах синтеза законов распределения вероятностей подтвердили эффективность предложенного подхода: алгоритм оптимизации с синтезированными законами распределения превзошел базовый метод L-SHADE на большинстве тестовых задач, особенно на задачах высокой размерности и сложности, обеспечив при этом лучшую стабильность результатов.

Заключение

В данном исследовании был предложен и оценен гибридный подход KAN-GP для символьной регрессии, объединяющий преимущества сетей Колмогорова-Арнольда и генетического программирования. Экспериментальные результаты демонстрируют, что гибридный подход превосходит как автономные методы KAN, так и ГП на различных тестовых задачах и реальных наборах данных. Проведенные эксперименты показали, что гибрид KAN-GP достигает более высокой символьной точности по сравнению со стандартными сетями KAN. Предложенный подход успешно преодолевает разрыв между эффективной многомерной аппроксимацией KAN и символьной точностью ГП.

Предложенный гибридный метод KAN-GP демонстрирует потенциал для создания универсальных метаэвристических алгоритмов оптимизации, а также может быть применен для разработки узкоспециализированных решений для конкретных классов оптимизационных задач.

В дальнейшем исследования будут сосредоточены на разработке новых активационных функций для KAN и создании адаптивной стратегии выбора наиболее подходящих функций для отдельных узлов. С практической точки зрения, предложенный гибридный подход предлагает эффективный инструмент для синтеза и идентификации вероятностных распределений для различных стохастических алгоритмов оптимизации, а в перспективе и для методов машинного обучения и других методов.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Del Ser J., Osaba E., Molina D., et al. Bio-inspired computation: where we stand and what's next. *Swarm and Evolutionary Computation.* 2019;48:220–250. https://doi.org/10.1016/j.swevo.2019.04.008
2. Stanovov V., Kazakovtsev L., Semenkin E. Hyper-heuristic approach for tuning parameter adaptation in differential evolution. *Axioms*. 2024;13(1):59. https://doi.org/10.3390/axioms13010059
3. Al-Dabbagh R.D., Neri F., Idris N., Baba M.S. Algorithmic design issues in adaptive differential evolution schemes: review and taxonomy. *Swarm and Evolutionary Computation*. 2018;43:284–311. https://doi.org/10.1016/j.swevo.2018.03.008
4. Makke N., Chawla S. Interpretable scientific discovery with symbolic regression: a review. *Artificial Intelligence Review*. 2024;57(1):2. https://doi.org/10.1007/s10462-023-10622-0
5. Liu Z., Wang Y., Vaidya S., et al. *KAN: Kolmogorov-Arnold networks.* arXiv. URL: https://arxiv.org/abs/2404.19756 [Accessed 1st November 2025].
6. Storn R., Price K. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization.* 1997;11(4):341–359. https://doi.org/10.1023/A:1008202821328

7. Das S., Suganthan P.N. Differential evolution: a survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation.* 2011;15(1):4–31. https://doi.org/10.1109/TEVC.2010.2059031
8. Tao S., Zhao R., Wang K., Gao Sh. *An efficient reconstructed differential evolution variant by some of the current state-of-the-art strategies for solving single objective bound constrained problems.* arXiv. URL: https://arxiv.org/abs/2404.16280 [Accessed 1st November 2025].
9. Tanabe R., Fukunaga A.S. Improving the search performance of SHADE using linear population size reduction. In: *2014 IEEE Congress on Evolutionary Computation (CEC), 06–11 July 2014, Beijing, China.* IEEE; 2014. P. 1658–1665. https://doi.org/10.1109/CEC.2014.6900380
10. Qin A.K., Suganthan P.N. Self-adaptive differential evolution algorithm for numerical optimization. In: *2005 IEEE Congress on Evolutionary Computation: Volume 2, 02–05 September 2005, Edinburgh, UK.* IEEE; 2005. P. 1785–1791. https://doi.org/10.1109/CEC.2005.1554904
11. Brest J., Greiner S., Boskovic B., Mernik M., Zumer V. Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. *IEEE Transactions on Evolutionary Computation*. 2006;10(6):646–657. https://doi.org/10.1109/TEVC.2006.872133
12. Zhang J., Sanderson A.C. JADE: adaptive differential evolution with optional external archive. *IEEE Transactions on Evolutionary Computation*. 2009;13(5):945–958. https://doi.org/10.1109/TEVC.2009.2014613
13. Tanabe R., Fukunaga A. Success-history based parameter adaptation for differential evolution. In: *2013 IEEE Congress on Evolutionary Computation, 20–23 June 2013, Cancun, Mexico*. IEEE; 2013. P. 71–78. https://doi.org/10.1109/CEC.2013.6557555
14. Das S., Mullick S.S., Suganthan P.N. Recent advances in differential evolution – an updated survey. *Swarm and Evolutionary Computation*. 2016;27(3). https://doi.org/10.1016/j.swevo.2016.01.004
15. Madala H.R., Ivakhnenko A.G. *Inductive learning algorithms for complex system modeling.* Boca Raton: CRC Press; 1993. 384 p.
16. Koza J.R. Genetic programming as a means for programming computers by natural selection. *Statistics and Computing.* 1994;4(2):87–112. https://doi.org/10.1007/BF00175355
17. Koza J.R. *Genetic programming II: automatic discovery of reusable programs.* Cambridge: MIT Press; 1994. 768 p.
18. Montana D.J. Strongly typed genetic programming. *Evolutionary Computation.* 1995;3(2):199–230. https://doi.org/10.1162/evco.1995.3.2.199
19. McKay R.I., Hoai N.X., Whigham P.A., Shan Y., O'Neill M. Grammar-based genetic programming: a survey. *Genetic Programming and Evolvable Machines.* 2010;11(3):365–396. https://doi.org/10.1007/s10710-010-9109-y
20. Cava W.L., Orzechowski P., Burlacu B., et al. *Contemporary symbolic regression methods and their relative performance.* arXiv. URL: https://arxiv.org/abs/2107.14351 [Accessed 1st November 2025].
21. Rimas M., Chen Q., Zhang M. Bloating reduction in symbolic regression through function frequency-based tree substitution in genetic programming. In: *AI 2023: Advances in Artificial Intelligence: 36th Australasian Joint Conference on Artificial Intelligence: Part II, 28 November – 01 December 2023, Brisbane, QLD, Australia*. Singapore: Springer; 2024. P. 429–440. https://doi.org/10.1007/978-981-99-8391-9_34

22. Liu Z., Ma P., Wang Y., Matusik W., Tegmark M. *KAN 2.0: Kolmogorov-Arnold networks meet science.* arXiv. URL: https://arxiv.org/abs/2408.10205 [Accessed 1st November 2025].
23. Somvanshi Sh., Javed S.A., Islam M.M., Pandit D., Das S. A survey on Kolmogorov-Arnold network. *ACM Computing Surveys.* 2025;58(2):55. https://doi.org/10.1145/3743128
24. Uy N.Q., Hoai N.X., O'Neill M., McKay R.I., Galván-López E. Semantically-based crossover in genetic programming: application to real-valued symbolic regression. *Genetic Programming and Evolvable Machines.* 2011;12(2):91–119. https://doi.org/10.1007/s10710-010-9121-2
25. Mundhenk T.N., Landajuela M., Glatt R., Santiago C.P., Faissol D.M., Petersen B.K. Symbolic regression via deep reinforcement learning enhanced genetic programming seeding. In: *Advances in Neural Information Processing Systems 34 (NeurIPS 2021), 06–14 December 2021, Virtual Event*. 2021. P. 24912–24923.

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Сопов Евгений Александрович, доктор технических наук, доцент, профессор, Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнёва, Красноярск, Российская Федерация.
e-mail: evgenysopov@gmail.com
ORCID: 0000-0003-4410-7996

Evgenii A. Sopov, Doctor of Engineering Sciences, Docent, Professor, Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, the Russian Federation.

Сопов Антон Евгеньевич, студент, Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнёва, Красноярск, Российская Федерация.
e-mail: antonsopov2004@gmail.com
ORCID: 0009-0002-3917-2283

Anton E. Sopov, Student, Reshetnev Siberian State University of Science and Technology, Krasnoyarsk, the Russian Federation.

Новиков Захар Борисович, аспирант, Сибирский федеральный университет, Красноярск, Российская Федерация.
e-mail: zakharzokhar@gmail.com
ORCID: 0009-0001-4512-2701

Zakhar B. Novikov, Postgraduate, Siberian Federal University, Krasnoyarsk, the Russian Federation.

Статья поступила в редакцию 11.12.2025; одобрена после рецензирования 23.10.2026; принята к публикации 23.11.2026.

The article was submitted 11.12.2025; approved after reviewing 23.10.2026; accepted for publication 23.11.2026.