



Получено: 27.01.2026 г. | Принято: 30.01.2026 г. | DOI: <https://doi.org/10.22184/1993-8578.2026.19.1.80.88>

Научная статья

# ОПЫТ ПРИМЕНЕНИЯ ОТКРЫТОЙ БИБЛИОТЕКИ ГРАДИЕНТНОГО БУСТИНГА CATBOOST НА ЭТАПЕ ТРАССИРОВКИ В РАМКАХ ФИЗИЧЕСКОГО СИНТЕЗА НА ОСНОВЕ ПЛИС

А.Ю.Чистяков<sup>1</sup>, инж.-иссл., ORCID: 0000-0002-5972-8317 / [chis.alexandr2018@yandex.ru](mailto:chis.alexandr2018@yandex.ru)  
М.А.Заплетина<sup>1</sup>, к.т.н., ст. науч. сотр., ORCID: 0000-0001-9845-7823

**Аннотация.** Разработана модель прогнозирования времени трассировки в основе программного модуля подбора входных параметров для ускорения выполнения этапа трассировки средствами САПР X-CAD для реализации цифровых схем на ПЛИС. Вычислительные эксперименты продемонстрировали сокращение времени генерации трассировочного решения в 1,5 раза в среднем для ПЛИС 5510ТС028.

**Ключевые слова:** трассировка, САПР X-CAD, ПЛИС, цифровые схемы

**Для цитирования:** А.Ю. Чистяков, М.А. Заплетина. Опыт применения открытой библиотеки градиентного бустинга catboost на этапе трассировки в рамках физического синтеза на основе ПЛИС. НАНОИНДУСТРИЯ. 2026. Т. 19. № 1. С. 80–88. <https://doi.org/10.22184/1993-8578.2026.19.1.80.88>.

Received: 27.01.2026 | Accepted: 30.01.2026 | DOI: <https://doi.org/10.22184/1993-8578.2026.19.1.80.88>

Original paper

# EXPERIENCE OF APPLICATION OF CATBOOST OPEN-SOURCE GRADIENT BOOSTING LIBRARY FOR ROUTING STAGE WITHIN FPGA PHYSICAL SYNTHESIS

A.Y.Chistiakov<sup>1</sup>, Engineer Researcher, ORCID: 0000-0002-5972-8317 / [chis.alexandr2018@yandex.ru](mailto:chis.alexandr2018@yandex.ru)  
M.A.Zapletina<sup>1</sup>, Cand. of Sci. (Tech), Senior Researcher, ORCID: 0000-0001-9845-7823

**Abstract.** A model for predicting routing time has been developed, forming the core of a software module for tuning input parameters to accelerate the routing stage using the X-CAD computer-aided design tool for implementing digital circuits on FPGAs. Computational experiments demonstrated a reduction of routing solution generation time by an average of 1.5 times for the 5510TS028 FPGA.

**Keywords:** routing, X-CAD computer-aided design, FPGA, digital circuits

**For citation:** T.O. Sovetnikov, V.M. Ganshin, I.V. Yaminsky. Experience of application of catboost open-source gradient boosting library for routing stage within FPGA physical synthesis. NANOINDUSTRY. 2026. Vol. 19. No. 1. PP. 80–88. <https://doi.org/10.22184/1993-8578.2026.19.1.80.88>.

<sup>1</sup> ОППМ ЦПМ НИЦ "Курчатовский институт", Москва, Россия / DDPM CPM NRC "Kurchatov Institute", Moscow, Russia



## ВВЕДЕНИЕ

В текущих условиях масштабного импортозамещения возникает необходимость все более широкого применения отечественных программируемых логических интегральных схем (ПЛИС). Ключевым фактором, оказывающим влияние на качество результатов проектирования в их базе, являются системы автоматизированного проектирования (САПР), их производительность и удобство работы с ними.

Маршрут проектирования схем в базе ПЛИС состоит из этапов логического синтеза, физического синтеза (включающего кластеризацию, размещение и трассировку), цифрового моделирования, статического временного анализа, создания прошивки. Наиболее затратным по времени и объему вычислений является предварительный этап трассировки в составе этапа физического синтеза. Кроме того, именно на нем значительным образом негативно отражается тенденция к росту степени интеграции новых микросхем. В связи с этим, его ускорение играет важную роль с точки зрения эффективности САПР.

Одним из возможных способов ускорения этапа трассировки является применение методов машинного обучения (МО). В настоящее время ведущие разработчики САПР для ПЛИС (компании Altera, AMD) уже получили первый успешный опыт внедрения интеллектуальных методик в своих продуктах, и исследования в этой области представляются наиболее перспективными.

В данной статье описан опыт применения открытой библиотеки машинного обучения CatBoost [1] для подбора параметров алгоритма трассировки в составе САПР для ПЛИС X-CAD [2]. В разделе 2 рассмотрены существующие подходы к применению методов машинного обучения в маршруте проектирования схем на ПЛИС. Раздел 3 посвящен использованным в данной работе инструментам. Раздел 4 содержит описание идеи применения модели CatBoost для этапа трассировки и ее реализации в составе маршрута проектирования в САПР X-CAD. В разделе 5 представлены результаты вычислительных экспериментов и анализ влияния разработанного подхода на итоговые характеристики проектируемых схем. В разделах 6 и 7 приведены некоторые трудности подхода, возникшие в ходе разработки, и заключение, соответственно.

## АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДИК ПРИМЕНЕНИЯ МАШИННОГО ОБУЧЕНИЯ В МАРШРУТЕ ПРОЕКТИРОВАНИЯ СХЕМ В БАЗИСЕ ПЛИС

Технологии машинного обучения, получившие стремительное развитие в последнее время, находят применение во множестве сфер. Исключением не стали

## INTRODUCTION

In the current conditions of large-scale import substitution, there is a need for increasingly widespread use of domestic programmable logic integrated circuits (FPGAs). A key factor influencing the design results quality based on them is computer-aided design (CAD) systems, their performance and ease of use.

The route for designing circuits in the FPGA database consists of the stages of logical synthesis, physical synthesis (including clustering, placement and routing), digital modelling, static timing analysis, and firmware creation. The most time-consuming and computationally intensive stage is the preliminary routing stage, which is part of the physical synthesis stage. In addition, it is this stage that is significantly affected by the trend towards greater integration of new microcircuits. In this regard, accelerating this stage plays an important role in terms of CAD efficiency.

One possible way to speed up the tracing stage is to use machine learning (ML) methods. Currently, leading developers of CAD for FPGAs (Altera, AMD) have already had their first successful experience of implementing intelligent techniques in their products, and research in this area appears to be the most promising.

This paper describes experience of using the CatBoost open machine learning library [1] to select tracing algorithm parameters in the X-CAD CAD system for FPGAs [2]. Section 2 reviews existing approaches to machine learning application methods in the FPGA circuit design flow. Section 3 is devoted to the tools used in this work. Section 4 describes the idea of applying the CatBoost model to the tracing stage and its implementation as part of the design route in the X-CAD CAD system. Section 5 presents the results of computational experiments and an analysis of the impact of the developed approach on the final characteristics of the designed circuits. Sections 6 and 7 describe some of the difficulties encountered during development and the conclusion, respectively.

## ANALYSIS OF EXISTING METHODS FOR APPLYING MACHINE LEARNING IN THE DESIGN PROCESS OF CIRCUITS IN THE FPGA DATABASE

Machine learning technologies, which have undergone rapid development in recent years, are used in a variety of fields. CAD for FPGAs is no exception, with machine learning methods being incorporated into design algorithms to significantly improve the quality of the resulting solutions. For example, AMD, the second largest developer in terms of sales on the global FPGA market, uses these technologies in its Vivado CAD [3]. Their use allows for more efficient logical optimisation, circuit delay estimation, and certain other operations. Collectively, the application of machine learning methods has made it possible to significantly speed up route traversal while simultaneously improving the quality of design results.



и САПР для ПЛИС, в которых методы машинного обучения внедряются в алгоритмы проектирования и позволяют значительно повысить качество получаемых итоговых решений. Так, данные технологии в работе САПР Vivado [3] применяет компания AMD, второй по объему реализуемой продукции разработчик на мировом рынке ПЛИС. Их использование позволяет эффективнее производить логическую оптимизацию, оценку задержек схемы и некоторые другие операции. В совокупности применение методов машинного обучения позволило кратно ускорить прохождение маршрута с одновременным улучшением качества результатов проектирования.

Активные усилия разработчиков направлены на применение методов машинного обучения для улучшения работы отдельных этапов проектирования, в частности – трассировки. Она представляет собой заключительный этап физического синтеза в маршруте проектирования схем на ПЛИС. В результате ее выполнения обеспечивается отображение проектных цепей, полученных на этапе логического синтеза, на имеющиеся трассировочные ресурсы ПЛИС. Таким образом, осуществляется соединение логических элементов и других компонентов ПЛИС (ячеек ввода/вывода, сложно-функциональных блоков) между собой. От качества выполнения трассировки напрямую зависят временные характеристики получаемых схем, поэтому улучшение применяемых в САПР алгоритмов трассировки остается важным направлением для исследований. Далее представлены актуальные работы в этой связи.

Одной из задач, решаемых в работе [4], является прогнозирование разводимости проектных цепей на основе данных предыдущих этапов проектирования. Основой предсказывающей модели DLRoute выступает сверточная нейронная сеть. В качестве входных данных она принимает четыре карты перегруженности, отражающие длину межсоединений на единицу площади, количество терминалов проектных цепей и количество разрезанных цепей (имеющих внешние терминалы) внутри двух различных по площади фрагментов поверхности кристалла. В результате анализа карт модель делает предсказание о разводимости схемы с таким размещением. Применение данного подхода позволило пропускать выполнение некоторых этапов проектирования для схем с хорошей разводимостью и тем самым ускорило выполнение маршрута на 48,3%. Суммарная длина межсоединений при этом увеличилась на 4,3%.

В статье [5] описана разработка генеративно-состязательной нейронной сети, с помощью которой на основе данных о размещении и взаимной

Developers are actively working to apply machine learning methods to improve individual design stages, particularly routing. Routing is the final stage of physical synthesis in the FPGA circuit design process. As a result, the design circuits obtained at the logic synthesis stage are mapped onto the available FPGA routing resources. This connects the logic elements and other FPGAs components (input/output cells, complex functional blocks) to each other. The timing characteristics of the resulting circuits depend directly on the quality of the routing, so improving the routing algorithms used in CAD remains an important area of research. Current work in this area is presented below.

One of the tasks addressed in [4] is predicting the separability of design circuits based on data from previous design stages. The DLRoute prediction model is based on a convolutional neural network. It takes four congestion maps as input data, reflecting the length of interconnections per unit area, the number of design circuit terminals, and the number of split circuits (having external terminals) within two different areas of the crystal surface. Based on the analysis of the maps, the model predicts the layout routability with this placement. Using this approach made it possible to skip certain design stages for layouts with good routability, thereby speeding up the routing process by 48.3%. The total length of interconnections increased by 4.3%.

The paper [5] describes development of a generative-competitive neural network, which was used to create a heat map of the potential location of overloaded areas of the crystal at the tracing stage, based on data on the placement and interconnection of FPGA elements. Based on the results of its analysis, changes can be made to the placement of the designed circuit to reduce the complexity of the routing task, which will ultimately lead to improved routability and higher quality of the resulting routing solution.

Another category of scientific work includes research on machine learning methods application at the top level in relation to software for designing integrated circuits, for example, for selecting CAD input parameters that improve the quality of design results. An example of such work is [6]. It describes the machine learning methods application for tuning the parameters of the algorithm for placing circuits that include complex functional blocks. On a number of test circuits, the proposed solution demonstrated the fundamental possibility of achieving improvements in both classic industry metrics of area, delay, and power consumption, as well as in the number of design rule violations (DRC) and total interconnect length.

## SOFTWARE USED

### 1. CatBoost Library

The basis for development a predictive model in this work was the open machine learning library CatBoost



связанности элементов ПЛИС создавалась тепловая карта потенциального расположения перегруженных участков кристалла на этапе трассировки. На основе результатов ее анализа могут быть внесены изменения в размещение проектируемой схемы для снижения сложности трассировочной задачи, что в итоге позволит достичь улучшенной разводимости и более высокого качества получаемого трассировочного решения.

К иной категории научных работ относятся исследования по применению методов машинного обучения на верхнем уровне по отношению к программному обеспечению для проектирования СБИС, например, для подбора входных параметров САПР, обеспечивающих улучшение качества результатов проектирования. Примером такой работы является статья [6]. В ней описано применение методов машинного обучения для настройки параметров алгоритма размещения схем, включающих сложно-функциональные блоки. На ряде тестовых схем предложенное решение показало принципиальную возможность достичь улучшения как по классическим отраслевым метрикам площади, задержки и потребляемой мощности, так и по количеству нарушений правил проектирования (DRC) и суммарной длине межсоединений.

## ИСПОЛЬЗУЕМЫЕ ПРОГРАММНЫЕ СРЕДСТВА

### 1. Библиотека CatBoost

Основой для создания прогнозирующей модели в данной работе выступила открытая библиотека машинного обучения CatBoost на языке программирования Python, разработанная компанией Яндекс. В основе формируемых на ее основе моделей находится метод градиентного бустинга. Он подразумевает объединение нескольких слабых алгоритмов машинного обучения в один сильный, где каждый последующий алгоритм учитывает ошибки предыдущего. CatBoost оптимизирована для работы с категориальными признаками, что позволяет использовать их без предварительной обработки. Библиотека применяется в работе многих сервисов компании Яндекс при выполнении задач по прогнозированию погоды, выдаче поисковых запросов и др.

Ключевыми факторами, повлиявшими на выбор CatBoost в качестве инструмента для получения опыта применения методов МО в САПР X-CAD, послужило следующее: библиотека является отечественной разработкой; модели, созданные с ее помощью, требуют меньше времени на обучение и быстрее проводят предсказание по сравнению с аналогами; для нее представлена качественная, подробная документация и множество примеров использования в открытом доступе.

in the Python programming language, developed by Yandex. The models formed on its basis are based on the gradient boosting method. This involves combining several weak machine learning algorithms into one strong one, where each subsequent algorithm takes into account the errors of the previous one. CatBoost is optimised to work with categorical features, which allows them to be used without pre-processing. The library is used in many Yandex services for weather forecasting, search queries, and other tasks.

The key factors that influenced the choice of CatBoost as a tool for gaining experience in applying ML methods in X-CAD CAD were as follows: the library is a domestic development; models created with its help require less training time and make predictions faster than their counterparts; it is accompanied by high-quality, detailed documentation and numerous publicly available examples of use.

### 2. Optuna framework

To develop a model with the highest prediction accuracy, the CatBoost library requires the configuration of model hyperparameters, such as the selection of the number of training iterations. Manual iterative selection of these parameters is possible, but this approach is not optimal because the relationship between hyperparameters and training results is not obvious. The Optuna framework [7] comes to the rescue in solving this problem by automating the selection of hyperparameters. This tool supports a set of optimisation and selection interruption algorithms, allowing the framework to be configured in the most effective way for a specific task. Optuna is also implemented in Python, which lowers the threshold for specialists to apply it with the CatBoost library.

## IDEA OF APPLICATION AND IMPLEMENTATION

### 1. The place of the model in the design process

Routing is one of the stages of the standard route for designing circuits on FPGAs, which is shown in Fig.1. In the X-CAD CAD system, it is performed using a modification of the PathFinder routing algorithm [8], which has a number of configuration parameters that control the direction and dynamics of the search for a routing solution. Preliminary experiments have shown that by selecting these parameters, it is possible to significantly reduce the routing time and improve the quality of the result.

By default, the parameter set values and tracing algorithm dynamics remain unchanged, regardless of the characteristics of the designed circuit and its placement on the FPGA chip. The idea behind using the CatBoost library is to create a predictive model for determining the characteristics of the tracing stage results based on circuit placement data, and then use the data obtained to select individual parameter set values

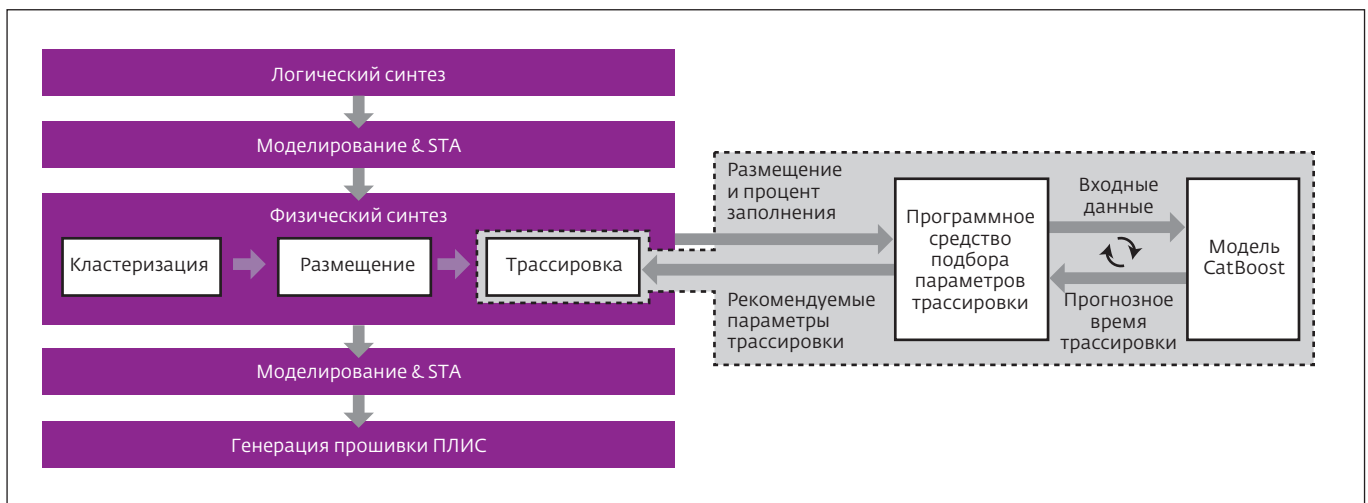


Рис.1. Схема применения модели CatBoost в маршруте проектирования X-CAD

Fig.1. Application scheme of CatBoost model in X-CAD design flow

## 2. Фреймворк Optuna

Для создания модели с наибольшей точностью предсказания библиотека CatBoost требует настройки гиперпараметров модели, например, подбора количества итераций обучения. Ручной итерационный выбор этих параметров возможен, однако такой подход не оптимален, поскольку зависимость между гиперпараметрами и результатом обучения не очевидна. На помощь в решении данной задачи приходит фреймворк Optuna [7], который автоматизирует подбор гиперпараметров. Этот инструмент поддерживает набор алгоритмов оптимизации и прерывания подбора, что позволяет настроить фреймворк наиболее эффективным образом для решения конкретной задачи. Optuna также реализован на языке Python, что снижает порог вхождения специалистов в задачу применения его с библиотекой CatBoost.

### ИДЕЯ ПРИМЕНЕНИЯ И РЕАЛИЗАЦИЯ

#### 1. Место модели в маршруте проектирования

Трассировка является одним из этапов стандартного маршрута проектирования схем на ПЛИС, который показан на рис.1. В САПР X-CAD она выполняется с помощью модификации алгоритма трассировки PathFinder [8], которая имеет ряд настроечных параметров, регулирующих направление и динамику поиска трассировочного решения. Предварительные эксперименты показали, что путем подбора этих параметров можно значительно уменьшать время трассировки, а также улучшать качество получаемого результата.

По умолчанию значения набора параметров и динамика алгоритма трассировки остаются неизменными, независимо от особенностей проектируемой схемы и ее размещения на кристалле ПЛИС.

for each circuit being designed. The proposed approach is designed to eliminate the labour-intensive manual selection of tuning parameters and speed up the final solution.

To implement the idea, an auxiliary software module was developed as part of the X-CAD CAD system, shown schematically in Fig.1 on the right and highlighted with a dotted line. The main structures included in it are a regression model based on the CatBoost library and a parameter selection tool that acts as a link between the model and the tracing module in the CAD software core.

From the previous stages of the route, the parameter selection tool receives information about the placement of the circuit (in matrix form) and the percentage of the FPGA crystal area filled. The tool then generates dozens of variations of the set of tuning parameters based on predefined ranges of values and step sizes. Once the parameter sets have been formed, the CatBoost model launch cycle begins. The number of iterations in the cycle is equal to the number of sets. During each iteration, one of the generated sets, the matrix layout of the circuit, and the crystal fill percentage are passed to the model, which uses this data to predict the trace time and returns it to the matching tool. At the end of the cycle, the obtained trace time predictions are compared with each other, and the set of parameters that yielded the shortest time is passed back to the route to control the tracing algorithm.

#### 2. Preparation of the data set

Training the CatBoost regression model requires a large dataset containing trace parameter values, placement, crystal fill percentage, and corresponding trace time. To develop such a dataset, circuits were



Идея применения библиотеки CatBoost заключается в формировании предсказательной модели для определения характеристик результата этапа трассировки на основании данных о размещении схемы, и в дальнейшем использовании полученных данных для подбора индивидуальных значений набора параметров для каждой проектируемой схемы. Предложенный подход призван устранить стадию трудоемкого ручного подбора настроечных параметров и ускорить получение итогового решения.

Для реализации идеи был разработан вспомогательный программный модуль в составе САПР X-CAD, схематично показанный на рис.1 справа и выделенный пунктирной линией. Основными структурами, входящими в его состав, являются регрессионная модель на основе библиотеки CatBoost и программное средство подбора параметров, выполняющее функцию связующего звена между моделью и модулем трассировки в программном ядре САПР.

Из предыдущих этапов маршрута программному средству подбора параметров поступает информация о размещении схемы (в матричном виде) и проценте заполнения площади кристалла ПЛИС. Затем средство генерирует десятки вариаций набора настроечных параметров на основе заранее заданных диапазонов значений и величины шага. После того, как наборы параметров сформированы, начинается цикл запусков модели CatBoost. Число итераций цикла равно количеству наборов. В ходе каждой итерации один из сгенерированных наборов, матричное размещение схемы и процент заполнения кристалла передаются модели, которая по этим данным прогнозирует время трассировки и возвращает его средству подбора. По завершении цикла полученные значения прогнозных времен трассировки сравниваются между собой, и набор параметров, при котором получено наименьшее время, передается назад в маршрут для управления алгоритмом трассировки.

## 2. Подготовка набора данных

Обучение регрессионной модели CatBoost требует большого набора данных, содержащего значения трассировочных параметров, размещение, процент заполнения кристалла и соответствующее им время трассировки. Для создания такого набора были отобраны схемы из состава открытых тестовых наборов LGSynth'89, IWLS'2005, ISCAS'85, ISCAS'89 и проектов с сайта opencores.org. В качестве целевого кристалла выбрана ПЛИС 5510TC028, имеющая 17920 программируемых логических элементов (ПЛЭ). Для схем выполнена начальная часть маршрута проектирования в X-CAD для генерации их размещения. Процент заполнения площади кристалла варьировался от 0,1 до 35%. На основе полученных вариантов

selected from the open test sets LGSynth'89, IWLS'2005, ISCAS'85, ISCAS'89, and projects from opencores.org. The target crystal selected is the 5510TS028 FPGA, which has 17,920 programmable logic elements (PLEs). For the circuits, the initial part of the design route was completed in X-CAD to generate their placement. The percentage of crystal area coverage varied from 0.1 to 35%. Based on the options obtained, the circuit tracing stage was launched with random variations in the tracing parameters. Thus, a training data set was formed, consisting of 2,000 records, each containing 5 parameters of the tracing algorithm [2]:

1. factor\_bt – directed search influence coefficient;
2. maxstep\_bt – depth of construction of reverse trees of directed search;
3. factor\_vp – coefficient of influence of tracing resource overload;
4. factor\_vh – coefficient of influence of tracing resource overload on previous iterations;
5. max\_path\_w – maximum weight of the tracing path.

The record also includes the crystal fill percentage, placement information in the form of binary value matrices (0 – free PLL, 1 – used in circuit placement), and the trace time obtained in seconds as target data.

## 3. Training models

The predictive model was trained on a prepared dataset using the CatBoostRegressor strategy. The dataset was automatically split into training and test sets using CatBoost tools, with a 90% to 10% ratio, respectively. Model hyperparameter tuning was carried out using the Optuna framework, utilizing the TPESampler optimization algorithm and the SuccessiveHalvingPruner early stopping algorithm during the process.

The selected hyperparameter values are as follows:

1. learning\_rate – 0,05;
2. iterations – 200;
3. depth – 8;
4. boosting\_type – Plain.

Placement information was used in the form of categorical data. The accuracy of the resulting model, assessed using the Mean Absolute Percentage Error (MAPE), was approximately 35%.

## 4. Development of auxiliary software tools

The routing parameter selection software is designed to automate the interaction of the X-CAD design route with a CatBoost-based predictive model and is a program written in Tcl. For each of the parameters required by the model to predict routing time, a range of values and a step for generating variations are specified. Due to the extremely large amount of time required to exhaustively go through all combinations of tracing parameters, the most suitable values for each parameter are calculated separately. Thus, five cycles



произведены запуски этапа трассировки схем со случайными вариациями трассировочных параметров. Таким образом, сформирован набор данных для обучения, состоящий из 2000 записей, каждая из которых содержит 5 параметров алгоритма трассировки [2]:

1. `factor_bt` – коэффициент влияния направленного поиска;
2. `maxster_bt` – глубина построения обратных деревьев направленного поиска;
3. `factor_up` – коэффициент влияния перегруженности трассировочного ресурса;
4. `factor_vh` – коэффициент влияния перегруженности трассировочного ресурса на предыдущих итерациях;
5. `max_path_w` – максимальное значение веса трассировочного пути.

Также запись включает процент заполнения кристалла, информацию о размещении в виде матриц бинарных значений (0 – свободный ПЛЭ, 1 – используемый в размещении схемы), а также полученное время трассировки в секундах в качестве целевых данных.

### 3. Обучение модели

Прогнозирующая модель обучена на подготовленном наборе данных с помощью стратегии CatBoostRegressor. Разделение набора на обучающую и тестовую выборки произведено автоматически с помощью средств CatBoost с соотношением данных 90% и 10%, соответственно. Подбор гиперпараметров модели проведен с помощью фреймворка Optuna, при работе которого использован алгоритм оптимизации TPESampler и алгоритм прерывания подбора SuccessiveHalvingPruner.

Подобранные значения гиперпараметров выглядят следующим образом:

1. `learning_rate` – 0,05;
2. `iterations` – 200;
3. `depth` – 8;
4. `boosting_type` – Plain.

Сведения о размещении использованы в виде категориальных данных. Точность получившейся модели, оцененная с помощью средней абсолютной процентной ошибки (MAPE) составила порядка 35%.

### 4. Разработка вспомогательных программных средств

Программное средство подбора параметров трассировки предназначено для автоматизации взаимодействия маршрута проектирования X-CAD с прогнозирующей моделью на основе CatBoost и представляет собой программу на языке Tcl. Для каждого из параметров, необходимых модели для предсказания времени трассировки, задается диапазон значений и шаг генерирования вариаций. Ввиду

are performed, in each of which one of the parameters is selected. Parameters that have not yet been selected are replaced with the values used by default at the tracing stage in X-CAD. Information about the FPGA utilization and placement is the same for all runs of the CatBoost model. The value of the tunable parameter for which the model predicted the shortest trace time is considered the most suitable. Once all parameters have been selected, they are passed to the tracing algorithm.

Due to the regression-based type of the predictive model, in the case of small schemes, there is a possibility of obtaining a negative predicted time. If this occurs, the negative value is replaced with the minimum positive value is 1 second.

### RESULTS OF COMPUTATIONAL EXPERIMENTS

Computational experiments conducted by running the X-CAD design flow for a set of 86 typical circuits on the target FPGA 5510TC028 showed a reduction in routing time by an average of 1.5 times for circuits with a utilization rate of more than 5% (Fig.2). For circuits with a utilization rate of less than 5%, some acceleration in routing was also observed; however, the routing time for them is comparable to the time required for the parameter selection stage, making the use of the developed tools impractical.

The critical path delay in the current model was not an optimization criterion in this study, which is why the value of this metric increased on average by 10.5% for circuits with an occupancy rate of more than 5%. The number of tracing resources used decreased on average by 1.5% for similar circuits.

### DISCUSSION

One of the main drawbacks of the approach considered is the need to carry out a time-consuming process of retraining the model after each significant update of the X-CAD tracing algorithm. In addition, forming a high-quality training dataset in the form of a large sample of test circuits that ensures sufficient variability of characteristics presents a certain challenge.

An important aspect characteristic of forming a training dataset for ML methods for FPGA design is taking into account the features of the target programmable integrated circuit architectures, including their quantitative characteristics. For instance, the 5510TC028 chip used for computational experiments in the current work contains 192 input-output cells, which significantly limits the possible selection of circuits for implementation based on it. As a result, one can observe a certain heterogeneity in training and test data sets in terms of the fill percentage. The number of schemes that provide a high fill percentage is significantly lower

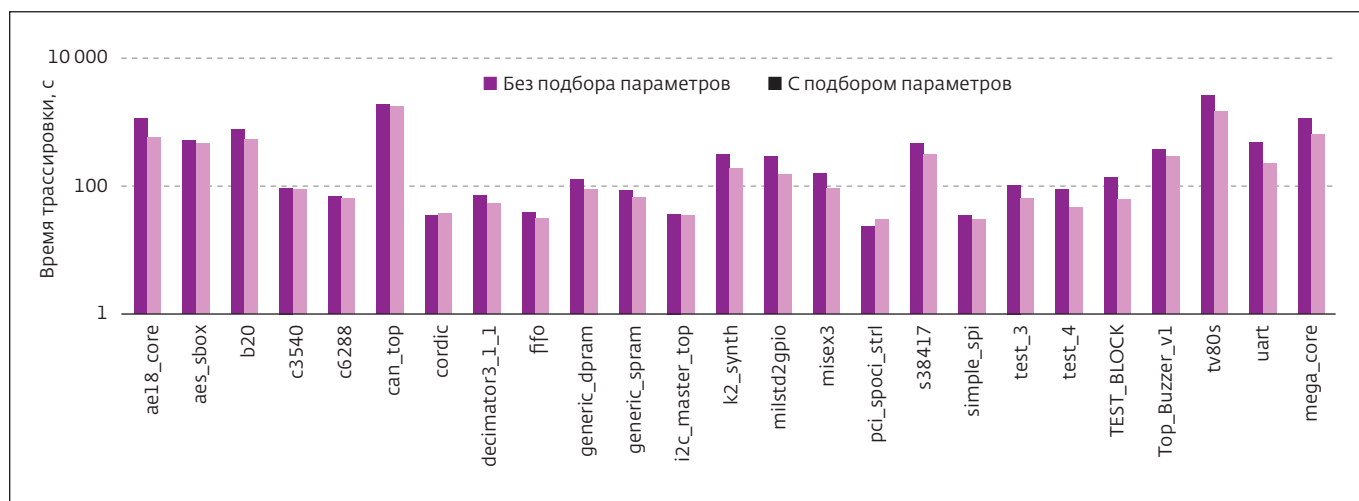


Рис.2. Влияние подбора трассировочных параметров на время трассировки некоторых типовых схем  
Fig.2. Influence of the tuning of routing parameters on the routing time of some typical circuits

крайне большого количества времени, требуемого на полный перебор всех комбинаций трассировочных параметров, наиболее подходящие значения для каждого параметра вычисляются отдельно. Таким образом, выполняются пять циклов, в каждом из которых происходит подбор одного из параметров. Параметры, подбор которых еще не проводился, замещаются значениями, используемыми на этапе трассировки в X-CAD по умолчанию. Данные о проценте заполнения ПЛИС и размещение одинаковы для всех запусков модели CatBoost. Значение подбираемого параметра, при котором моделью было предсказано наименьшее время трассировки, признается наиболее подходящим. После того, как подбор всех параметров завершен, они поступают алгоритму трассировки.

В связи с регрессионным типом прогнозирующей модели, в случае малых схем существует вероятность получения отрицательного предсказанного времени. При возникновении такой ситуации отрицательное значение заменяется на минимальное положительное – 1 секунда.

### РЕЗУЛЬТАТЫ ВЫЧИСЛИТЕЛЬНЫХ ЭКСПЕРИМЕНТОВ

Вычислительные эксперименты, проведенные в виде запуска маршрута проектирования X-CAD для набора из 86 типовых схем на целевой ПЛИС 5510TC028, показали уменьшение времени трассировки в среднем в 1,5 раза для схем с долей заполнения более 5% (рис.2). Для схем с долей заполнения менее 5% также зафиксировано некоторое ускорение трассировки, однако время трассировки для них сопоставимо с временем выполнения этапа подбора параметров, что делает применение разработанных средств нецелесообразным.

than the number of small schemes in the set, which is presumably the reason for the low accuracy in predicting trace time.

In addition, the methods of representing placement data, which would allow the model to use them as efficiently as possible, require refinement and in-depth study. The method using binary value matrices does not account for the relationships between logical cells, and in the case of designing small circuits, placement in this form represents a highly sparse matrix.

### CONCLUSIONS

The paper presents an approach to selecting parameters for a tracing algorithm using a predictive model based on the open-source machine learning library CatBoost. A dataset was created based on the results of X-CAD CAD runs, and a regression model was trained. The model's hyperparameters were tuned using the Optuna framework. Based on the results of computational experiments, the proposed approach demonstrated a reduction in routing time by an average of 1.5 times for designs with a target FPGA 5510TC028 utilization of over 5%. The critical path delay increased by an average of 10.5%, while the number of routing resources used decreased by an average of 1.5% for similar designs.

### PEER REVIEW INFO

Editorial board thanks the anonymous reviewer(s) for their contribution to the peer review of this work. It is also grateful for their consent to publish papers on the journal's website and SEL eLibrary eLIBRARY.RU.

**Declaration of Competing Interest.** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.



Время задержки критического пути в текущей модели не являлось критерием оптимизации в данной работе, в связи с чем значение этой метрики увеличилось в среднем на 10,5% для схем с долей заполнения более 5%. Количество использованных трассировочных ресурсов уменьшилось в среднем на 1,5% для аналогичных схем.

### ОБСУЖДЕНИЕ

Одним из основных недостатков рассмотренного подхода является необходимость проведения затратного по времени процесса переобучения модели после каждого значительного обновления алгоритма трассировки X-CAD. Помимо этого, определенную сложность составляет формирование качественного обучающего набора данных в виде большой выборки тестовых схем, обеспечивающих достаточную вариативность характеристик.

Важным аспектом, характерным для формирования обучающей выборки для методов МО для проектирования ПЛИС, является учет особенностей архитектуры целевых программируемых интегральных схем, в том числе их количественных характеристик. Так, схема 5510TC028, примененная для проведения вычислительных экспериментов в текущей работе, содержит 192 ячейки ввода-вывода, что значительным образом ограничивает возможный выбор схем для реализации на ее основе. Как следствие, можно наблюдать определенную неоднородность набора обучающихся и проверочных данных с точки зрения процента заполнения. Количество схем, обеспечивающих большой процент заполнения, значительно уступает количеству малых схем в наборе, что, предположительно, является причиной невысокой точности предсказания времени трассировки.

Кроме того, доработки и углубленного изучения требуют способы представления данных о размещении, которые позволило бы модели использовать их максимально эффективным образом. Способ с использованием матриц бинарных значений не позволяет учитывать связи между логическими ячейками, а в случае проектирования малых схем размещение в таком виде представляет собой крайне разреженную матрицу.

### ВЫВОДЫ

В работе представлен подход к подбору параметров алгоритма трассировки с использованием прогнозирующей модели на основе открытой библиотеки машинного обучения CatBoost. Создан набор данных на основе результатов запусков САПР X-CAD, и проведено обучение регрессионной модели. Настройка гиперпараметров модели осуществлена с помощью фреймворка Optuna. По результатам вычислительных экспериментов предложенный подход

продемонстрировал уменьшение времени трассировки в среднем в 1,5 раза для схем с долей заполнения целевого ПЛИС 5510TC028 более 5%. Время задержки критического пути увеличилось в среднем на 10,5%, а количество использованных трассировочных ресурсов уменьшилось в среднем на 1,5% для аналогичных схем.

### ИНФОРМАЦИЯ О РЕЦЕНЗИРОВАНИИ

Редакция благодарит анонимного рецензента (рецензентов) за их вклад в рецензирование этой работы, а также за размещение статей на сайте журнала и передачу их в электронном виде в НЭБ eLIBRARY.RU.

*Декларация о конфликте интересов. Авторы заявляют об отсутствии конфликтов интересов или личных отношений, которые могли бы повлиять на работу, представленную в данной статье.*

### ЛИТЕРАТУРА / REFERENCES

1. CatBoost – open-source gradient boosting library: <https://catboost.ai> (дата обращения: 07.07.2025).
2. Гаврилов С.В., Железников Д.А., Заплетина М.А., Тиунов И.В., Хватов В.М., Чочаев Р.Ж., Шокарев Д.Б. Разработка доверенных средств проектирования ИС в базе гетерогенных ПЛИС. Труды ИСП РАН. 2023. Т. 35. № 5. С. 107–126.
3. Introducing Vivado ML (WIP): Презентация – URL: [https://www.amd.com/content/dam/xilinx/publications/presentations/vivado\\_ml\\_launch\\_media-presentation.pdf](https://www.amd.com/content/dam/xilinx/publications/presentations/vivado_ml_launch_media-presentation.pdf) (дата обращения: 07.07.2025).
4. Szentimrey H., Al-Hyari A., Foxcroft J., Martin T., Noel D., Grewal G., Areibi S. Machine Learning for Congestion Management and Routability Prediction within FPGA Placement. ACM Transactions on Design Automation of Electronic Systems (TODAES). 2020. Vol. 25. No. 5. PP. 1–25.
5. Yu C., Zhang Z. Painting on Placement: Forecasting Routing Congestion using Conditional Generative Adversarial Nets. Proceedings of the 56th Annual Design Automation Conference/ 2019. PP. 1–6.
6. Agnesina A., Rajvanshi P., Yang T., Pradipta G., Jiao A., Keller B., Khailany B., Ren H. AutoDMP: Automated DREAMPlace-based Macro Placement. Proceedings of the 2023 International Symposium on Physical Design (ISPD '23). Association for Computing Machinery. 2023. New York, USA. PP. 149–157.
7. Optuna – A hyperparameter optimization framework: <https://optuna.org/> (дата обращения: 07.07.2025).
8. McMurchie L., Ebeling C. PathFinder: A negotiation-based performance-driven router for FPGAs. Proceedings of the 3rd International ACM Symposium on FPGAs. 1995. Napa Valley, CA. PP. 111–117.



**ТЕХНОСФЕРА**  
РЕКЛАМНО-ИЗДАТЕЛЬСКИЙ ЦЕНТР

**100% ГАРАНТИЯ**  
ПОЛУЧЕНИЯ ВСЕХ НОМЕРОВ



Стоимость 2200 р. за номер  
Периодичность: 10 номеров в год  
[www.electronics.ru](http://www.electronics.ru)



Стоимость 1450 р. за номер  
Периодичность: 8 номеров в год  
[www.photonics.ru](http://www.photonics.ru)



Стоимость 1450 р. за номер  
Периодичность: 6 номеров в год  
[www.j-analytics.ru](http://www.j-analytics.ru)

# ПОДПИСКА НА ЖУРНАЛЫ

[www.technosphere.ru](http://www.technosphere.ru)



Стоимость 1300 р. за номер  
Периодичность: 8 номеров в год  
[www.lastmile.ru](http://www.lastmile.ru)



Стоимость 1300 р. за номер  
Периодичность: 8 номеров в год  
[www.nanoindustry.ru](http://www.nanoindustry.ru)



Стоимость 1800 р. за номер  
Периодичность: 4 номера в год  
[www.stankoinstrument.ru](http://www.stankoinstrument.ru)