

Нейросетевая система ассистированного преобразования сигналов для задач с плохо формализуемыми метриками качества

Н. С. Васильев

*Высшая школа компьютерных технологий и информационных систем
Санкт-Петербургский политехнический университет Петра Великого
vasiliev2.ns@edu.spbstu.ru*

К. К. Семенов

*Высшая школа компьютерных технологий и информационных систем
Санкт-Петербургский политехнический университет Петра Великого
semenov_kk@spbstu.ru*

Аннотация. В статье рассмотрена задача ассистированной настройки параметров алгоритмов обработки сигналов в условиях отсутствия формализуемой метрики качества результата. Обоснована необходимость включения человека в контур работы соответствующего программного обеспечения. Предложена архитектура интерактивной системы-ассистента с разделением вычислительного и клиентского уровней, поддержкой инкрементальных вычислений и онлайн-обучения нейросетевой модели. Реализован прототип на стеке HTML/JavaScript/Pyodide, обеспечивающий выполнение пользовательских Python-алгоритмов в изолированной среде браузера. Показано, что предложенный подход позволяет ускорить достижение удовлетворительного результата за счет реактивной визуальной обратной связи и интеллектуального предсказания параметров.

Ключевые слова: некорректные задачи, обработка сигналов, нейронные сети, человеко-машинное взаимодействие, плохо формализуемые метрики качества

I. ВВЕДЕНИЕ

Для значительной части прикладных задач цифровой обработки сигналов, являющимися некорректными или плохо поставленными, характерно отсутствие формализуемого критерия оценки качества получаемого результата, что приводит к сложностям или вовсе невозможности применения численных методов оптимизации. К таким задачам относят разделение перекрывающихся хроматографических пиков, восстановление пропущенных фрагментов временных рядов, выполнение обратной фильтрации и деконволюцию спектральных данных [1, 2]. Общей чертой перечисленных задач является то, что результат для них зачастую оценивается экспертным образом – на основе визуального анализа или интуитивного представления пользователя о корректности решения.

В настоящее время подобные задачи решают преимущественно с использованием заранее определенных алгоритмов, параметры которых подбирают вручную. На практике это приводит к необходимости многократного запуска пользовательских программ с различными конфигурациями и субъективной оценки результата, что значительно снижает эффективность работы. Широко распространенные программные средства, как правило, не предоставляют средств для ручной или эвристической

настройки параметров в реальном времени: специализированные программные средства для перечисленных выше задач – такие, как OriginLab и Chromeleon, – ориентированы на задачи с конкретно заданными метриками и не обеспечивают реактивной интерактивной настройки параметров в задачах экспертной оценки [3]. Таким образом, в настоящее время отсутствует универсальный инструмент для взаимодействия с пользовательскими алгоритмами без формализованных метрик качества результата с возможностью накопления и использования в дальнейшем пользовательского опыта, выражаемого в процессе определения настроек соответствующих реализацией численных методов. Данное обстоятельство указывает на потребность в специальном интерактивном программном средстве.

В данной работе предлагается архитектура системы-ассистента, сочетающей интерактивный пользовательский интерфейс, инкрементальные вычисления и нейросетевой модуль предсказания параметров, обучаемый на истории взаимодействий. Подход вписывается в известную парадигму «человек в контуре» (HITL) [4]. Представлена нейросетевая система, обеспечивающая ассистированную настройку параметров пользовательских алгоритмов обработки сигналов в задачах, для которых невозможно или затруднительно задать формализуемую метрику качества результата, допускающую ее численную оптимизацию, а итоговое решение о достижении поставленной цели осуществляется пользователем на основе визуального анализа и интуитивного представления о корректности полученного решения.

II. ОБЗОР ЛИТЕРАТУРЫ

Задача цифровой обработки данных является корректной по Адамару, если имеют место существование, единственность и устойчивость решения. Классические методы регуляризации некорректных задач – метод Тихонова, итерационные схемы – предполагают наличие явной целевой функции, что ограничивает их применимость в задачах субъективной оценки [1]. В литературе отмечается, что зачастую количественные метрики плохо коррелируют с воспринимаемым пользователем качеством результата [5]. Данное обстоятельство характерно для основных классов некорректных задач: для обратных задач (включающих в восстановление скрытых причин по следствиям) и для задач реконструкции (интерполяция

Исследование выполнено за счет гранта Российского научного фонда № 25-21-20109, <https://rscf.ru/project/25-21-20109/>, и средств гранта Санкт-Петербургского научного фонда № 25-21-20109.

пропусков и устранение искажений) решение, как правило, не определяется однозначно без привлечения экспертного мнения [2].

Концепция активного обучения [6] привлекает стратегии запроса экспертной разметки наиболее информативных примеров. Обзор HITL-подходов [4] выделяет три сценария: аннотирование данных, проверку гипотез и интерактивное управление моделью. Машинное обучение для задач хроматографии [7] показывает, что глубокое обучение достигает высокого качества разделения пиков, однако по-прежнему требует экспертной валидации. Инкрементальные методы и онлайн-обновление моделей рассмотрены в [8, 9].

III. АНАЛИЗ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Анализ наиболее распространённых программных систем для обработки сигналов указывает на необходимость разработки специализированного решения, которое обладало бы следующими свойствами: поддержка пользовательских алгоритмов, визуальная настройка их параметров, реактивность обратной связи, накопление пользовательского опыта, ориентация на экспертную интерпретацию. Несоответствие данным требованиям является характерной особенностью основных используемых в настоящее время программных средств.

1) Matlab Signal Processing Toolbox [10] предоставляет широкие возможности для обработки сигналов: пользователи могут составлять собственные скрипты обработки, что делает платформу гибким инструментом. Вместе ориентация интерфейса данного пакета на скриптовую модель приводит к тому, что параметры используемых алгоритмов задаются вручную, пользователь вынужден ожидать результата после запуска скрипта. Визуальное управление параметрами реализуется только при самостоятельном создании пользователем GUI. Matlab не накапливает историю взаимодействий и принципиально ориентирован на работу с формальными метриками.

2) OriginLab [11] обеспечивает визуализацию и анализ спектроскопических данных через графический интерфейс. Встроенная поддержка анализа пиков приводит к широкому применению данного программного средства в задачах хроматографии. Вместе с тем управление параметрами обработки является пошаговым: каждое изменение значения требует закрытия диалогового окна и выполнения нового расчета, что препятствует интуитивно понятной настройке. Система не ведет истории конфигураций и не адаптируется к конкретному пользователю.

3) Chromeleon [12] является узкоспециализированной платформой для хроматографического анализа. Визуальные инструменты анализа ориентированы на стандартизированные рабочие процессы. Подключение пользовательских алгоритмов невозможно, взаимодействие с параметрами требует ручного подтверждения при каждом изменении. Адаптация к пользователю отсутствует.

Как видно, ни один из рассмотренных программных пакетов не удовлетворяет отмеченным выше требованиям к решению задач с неформализуемой метрикой качества результата, что подчеркивает необходимость в создании специализированных средств.

IV. ПОСТАНОВКА ЗАДАЧИ

Пусть $\mathbf{x} \in R_n$ представляет собой вектор входных данных, $f(\mathbf{x}, \boldsymbol{\theta})$ – функция, задающая пользовательский алгоритм его обработки с вектором параметров $\boldsymbol{\theta} \in \Theta \subseteq R_m$. В классической формальной постановке задача подбора параметров состоит в численной оптимизации $\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} Q(f(\mathbf{x}, \boldsymbol{\theta}))$, где Q – заданный функционал качества. Для рассматриваемой в работе проблемы явный вид Q не может быть задан аналитически, поскольку определяется субъективной оценкой эксперта. Следовательно, традиционная численная оптимизация неприменима. Вместе с тем, программное средство, обеспечивающее присутствие «человека в контуре», должно аналогично итерационным схемам поиска оптимума, позволять непрерывное построение последовательности приближений $\boldsymbol{\theta}_1^*, \boldsymbol{\theta}_2^*, \dots, \boldsymbol{\theta}_k^*$, момент остановки для которого определяется пользователем. Подходы HITL указывают на целесообразность конформного отображения значений области Θ на 1D или 2D многообразии, в качестве которых обычно выступают отрезки либо плоскости. Если размерность m множества Θ оказывается большей двух, следует применять номограммы с дискретизацией Θ по части измерений. Теоретической основой соответствующих построений является известная теорема Колмогорова-Арнольда [13] о том, что каждая непрерывная функция нескольких переменных может быть представлена в виде суперпозиции непрерывных функций одной переменной и бинарной операции сложения.

Среди типовых прикладных задач, для которых характерна описанная ситуация следует выделить задачи обработки данных для аналитических измерений:

- при разделении хроматографических пиков среди множества допустимых значений $\boldsymbol{\theta}^*$ лишь часть является физически обоснованной, что по большому счету может быть валидировано лишь экспертом [7],
- при восстановлении пропусков в спектрах при отсутствии априорной информации множество решений в форме значений $\boldsymbol{\theta}^*$ является бесконечным, а выбор метода и его параметров определяется визуальным восприятием результата,
- при выполнении обратной фильтрации и деконволюции использование классических метрик качества (среднеквадратического отклонения, отношения сигнал/шум) приводит к избыточному усилению шума, а визуально приемлемый результат нередко сопряжен с большими погрешностями [5].

V. ПРЕДЛАГАЕМАЯ АРХИТЕКТУРА ПРОГРАММЫ-АССИСТЕНТА

Предлагаемая программная система-ассистент реализована в виде веб-приложения с разделением клиентского и вычислительного уровней (рис. 1). Клиентская часть обеспечивает визуализацию результатов, управление параметрами и накопление истории взаимодействий, а вычислительный модуль выполняет пользовательские алгоритмы и обучает нейросетевую модель. Взаимодействие уровней осуществляется через REST API и WebSocket.

Пользовательский интерфейс реализован в виде одностороннего приложения SPA на базе HTML/SCSS/JavaScript. Интерфейс построен по принципу пошагового взаимодействия: каждый шаг представлен отдельной секцией, последовательно активируемой пользователем. Реактивность обеспечивается средствами Chart.js [14] – библиотеки построения интерактивных графиков, поддерживающей перерисовку при каждом изменении параметра. В расширенной версии клиентская часть разработана на React [15], что обеспечивает компонентную архитектуру и виртуальный DOM для высокой производительности.

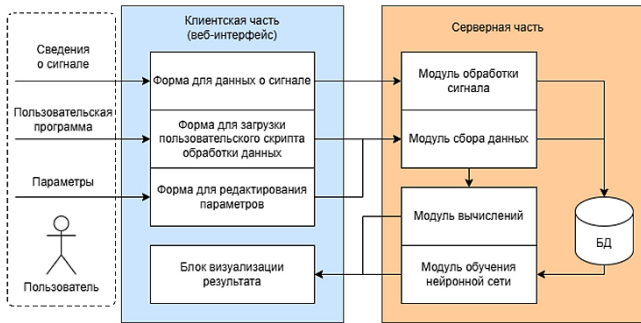


Рис. 1. Архитектура системы: клиентский и серверный уровни

Принципиальной особенностью системы является поддержка произвольных Python-алгоритмов пользователя для чего применен Pyodide [16] – интерпретатор Python, скомпилированный в WebAssembly и исполняющийся непосредственно в браузере. Это обеспечивает поддержку основных библиотек данного языка – NumPy, SciPy, Pandas – без серверной зависимости, полную изоляцию кода в браузерной «песочнице», максимальный отклик при изменении параметров. В масштабируемой архитектуре вычисления переносятся на сервер посредством Python/FastAPI [17], что снимает ограничения производительности.

Промежуточные пользовательские результаты, заключающиеся в получении приближений $\theta_1^*, \theta_2^*, \dots, \theta_k^*$ кэшируются на сервере; при изменении параметров осуществляется пересчет только затронутых данным изменением участков данных [8]. Это позволяет минимизировать объем вычислений при больших массивах данных и обеспечить субъективную мгновенность отклика на пользовательские действия по подбору целевых параметров используемых алгоритмов цифровой обработки сигналов.

Адаптация к предпочтениям пользователя обеспечивается нейросетевым модулем, обучаемым в режиме онлайн [9] на истории пар «состояние данных – выбранные параметры». Полносвязная сеть с активациями ReLU [18] обучается методами стохастической оптимизации [19]. После достаточного числа взаимодействий модуль предлагает начальные значения параметров, снижая когнитивную нагрузку на эксперта.

VI. РЕАЛИЗАЦИЯ ПРОГРАММНОГО ПРОТОТИПА

Прототип описываемой программной системы-ассистента реализован как клиентское веб-приложение без зависимости от серверной части. Минимальная архитектура включает три ключевых JavaScript-модуля:

1) основного управляющего модуля sidebar.js, осуществляющего переходы между шагами настройки (суть получение элементов последовательности $\theta_1^*, \theta_2^*, \dots, \theta_k^*$), загрузку файлов с данными, формирование вызова Python-функций;

2) модуля визуализации chart.js на основе Chart.js;

3) модуля pyRunner.js, осуществляющего взаимодействие с интерпретатором Pyodide, в частности инициализацию, передачу кода, обработку результата.

Взаимодействие модулей реализовано по принципу однонаправленного потока данных.

Шаг 1. Пользователь загружает csv-файл с данными.

Шаг 2. Пользователь загружает скрипт Python, реализующий алгоритм обработки данных с плохо формализуемой метрикой качества. Система разбирает скрипт регулярным выражением, извлекая сигнатуры функций и их аргументы.

Шаг 3. Пользователь выбирает функцию и задает диапазоны возможных значений параметров.

Шаг 4. Входной файл записывается во внутреннюю файловую систему Pyodide, происходит исполнение скрипта, результат в виде pandas.DataFrame конвертируется в файл формата csv и возвращается в JavaScript для визуализации.

Интерфейс настройки параметров должен обладать следующими основными элементами. Форма ввода параметров должна формироваться динамически на основе сигнатуры выбранной функции (рис. 2). Для элементов θ_{ij} , $j = 1, 2, \dots, m$, векторов подбираемой настройки θ_i^* предусмотрено задание диапазона возможных значений в формате $J_j = [\min, \max]$, где \min и \max – наименьшее и наибольшее допустимые значения параметра, задаваемые пользователем на основе экспертной информации или физических ограничений задачи. Данная процедура не выходит за рамки типовых настроек численных оптимизаторов и тем самым не налагает на пользователя требований наличия дополнительной априорной информации.

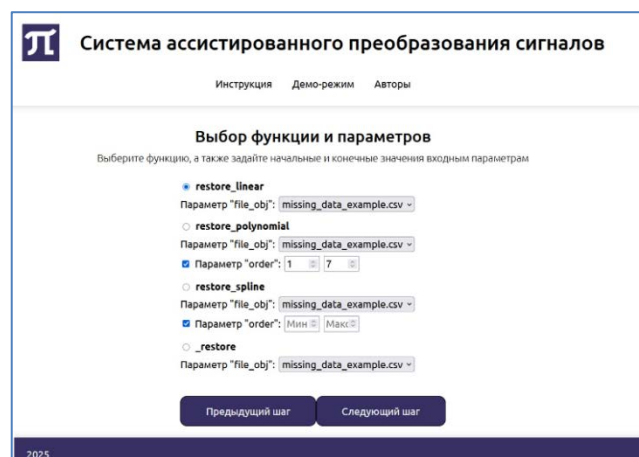


Рис. 2. Иллюстрация динамической формы для выбора загруженной пользовательской функции

На основе полученных диапазонов J_j формируется область взаимодействия с пользователем (ползунок либо прямоугольная область с перемещением курсора внутри нее). Управление активным указателем осуществляется

через компьютерную мышь или тачпад, изменение положения курсора приводит к автоматическому вызову пользовательской функции при измененном значении параметров θ и немедленному обновлению графического отображения результата цифровой обработки, что и обеспечивает реактивную визуальную обратную связь.

VII. ДЕМОСТРАЦИОННЫЕ ПРИМЕРЫ

Представленная нейросетевая система-ассистент апробирована на ряде задач, среди которых – задача разделения пиков в сигнале, полученном при электрофоретическом разделении нуклеиновых кислот, и задача восстановления пропущенных значений во временном ряду.

Для первой из задач параметрами θ выступали ширина аппаратной функции электрофореза и уровень случайной погрешности полученного сигнала. Пользовательский алгоритм цифровой обработки представлял собой код из [20, 21] для деконволюции на основе принципа минимального модуля и метода Говарда для подавления возникновения эффекта Гиббса. Так как решаемая задача естественным образом проецировалась на двумерную область, то $m = 2$ и в качестве области взаимодействия с пользователем выступала прямоугольная область, пределы которой определялись интервалами J_1 и J_2 соответственно. Пример восстановленного сигнала, построенный по полученному файлу в формате csv, представлен на рис. 3.

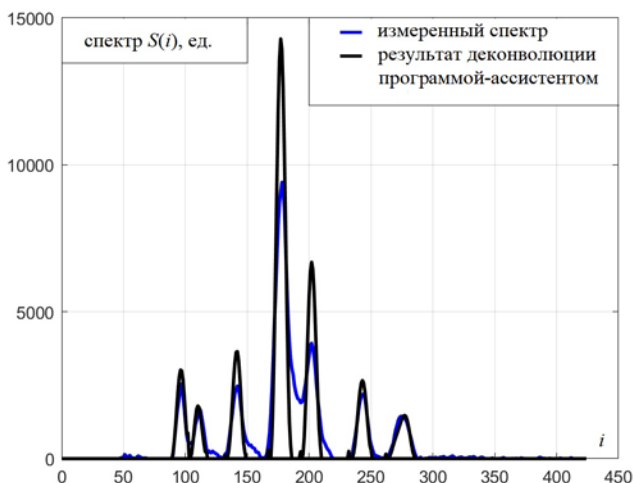


Рис. 3. Иллюстрация динамической формы для выбора загруженной пользовательской функции

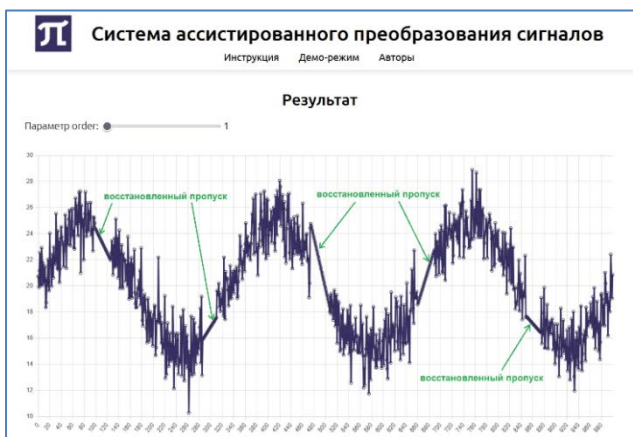


Рис. 4. Реактивный ползунок и восстановленный сигнал

Для второй задачи пользовательская программа обработки данных заключалась в заполнении пропусков во временной ряду методом наименьших квадратов, примененному к доступным данным, с заданным видом регрессионной зависимости. Подбирался всего один параметр зависимости, который изменялся непрерывно в области, определяемой интервалом J_1 . Соответственно взаимодействие с пользователем осуществлялось через ползунок на отрезке. Пример полученного результата в пользовательском окне разработанного программного средства представлен на рис. 4.

VIII. ТЕСТИРОВАНИЕ

Для представленного программного средства для ассистирования при решении задач обработки данных с плохо формализуемыми метриками качества результата были выполнены три вида тестирования.

- 1) Статический анализ с помощью ESLint выявил и устранил синтаксические конфликты, связанные с использованием ES-модулей.
- 2) Модульное тестирование средствами Jest и jsdom подтвердило корректность ключевых сценариев: загрузки файлов, переходов между шагами, обновления графика и обработки параметров.
- 3) Динамический анализ в Firefox Developer Tools зафиксировал равномерное распределение нагрузки, отсутствие утечек памяти и стабильную реактивность интерфейса при повторных вычислениях.

Предложенный в данной работе подход к решению некорректных задач принципиально отличается от традиционной автоматизированной обработки: он нацелен не на минимизацию целевой функции, а на ассистирование эксперту. Среди ограничений: зависимость предсказаний нейросетевого модуля от объема накопленной истории; ограниченность клиентских вычислений (Pyodide) для ресурсоёмких алгоритмов; ориентация на одного пользователя без механизмов переноса знаний. Авторы видят в качестве перспективных направлений развития представленного программного обеспечения переход к клиент-серверной архитектуре с вынесением вычислений на сервер, коллаборативный режим, интерпретируемость предсказаний, расширение поддерживаемых форматов данных.

IX. ЗАКЛЮЧЕНИЕ

В статье представлена архитектура и реализация системы-ассистента для задач обработки сигналов с плохо формализуемыми метриками качества. Анализ аналогов показал, что существующие системы для решения таких задач в интересах индустрии не удовлетворяют одновременно требованиям реактивности, произвольности выполняемых алгоритмов обработки и адаптации к пользователю. Ключевые особенности разработанного решения заключаются в реактивности интерфейса с немедленной визуальной обратной связью; в поддержке произвольных пользовательских Python-алгоритмов через изолированную среду Pyodide; в инкрементальном характере вычислений; в наличии нейросетевого модуля онлайн-обучения для ускорения решения повторяющихся задач. Программное средство апробировано на некорректных задачах цифровой

обработки данных. Область применения охватывает аналитическую химию, анализ спектроскопических данных, визуализацию медицинских и промышленных данных.

СПИСОК ЛИТЕРАТУРЫ

- [1] Бакушинский А.Б., Гончарский А.В. Некорректные задачи. Численные методы и приложения. М.: изд-во МГУ, 1989. 199 с.
- [2] Грешилов А.А. Некорректные задачи цифровой обработки информации и сигналов. М.: Логос, 2009. 358 с.
- [3] Цымблер М.Л., Юртин А.А. Восстановление пропущенных значений временного ряда на основе совместного применения аналитических алгоритмов и нейронных сетей // Вычислительные методы и программирование. 2023. Т.24. В.3. С. 243-259.
- [4] Wu, X., Xiao, L., Sun, Y., Zhang, J., Ma, T., He, L. (2022). A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems*, 135, 364-381.
- [5] Bertero, M., Boccacci, P., De Mol, C. (2021). *Introduction to Inverse Problems in Imaging* (2nd ed.) – Boca Raton, CRC Press. – 358 p.
- [6] Settles, B. (2010). *Active Learning Literature Survey*. Computer Sciences Technical Report 1648. University of Wisconsin-Madison. – 67 p.
- [7] Bosten, E., Chen, K., Hellings, M., Cabooter, D. (2025). Artificial intelligence for method development in liquid chromatography. *TrAC Trends in Analytical Chemistry*, 192, 118320.
- [8] Liu, Y.A. (2025). Incremental computation: what is the essence? *Foundations and Trends in Programming Languages*, 9(3), 178-244.
- [9] Ade, R.R., Deshmukh, P.R. (2013). Methods for incremental learning: a survey. *International Journal of Data Mining & Knowledge Management Process*, 3(4), 119.
- [10] MATLAB Signal Processing Toolbox [Электронный ресурс]. – URL: <https://www.mathworks.com/help/signal/index.html>
- [11] OriginLab [Электронный ресурс]. – URL: <https://www.originlab.com/index.aspx?go=Products/Origin>
- [12] Chromeleon Chromatography Data System (CDS) [Электронный ресурс]. – URL: <https://www.thermofisher.com/order/catalog/product/CHROMELEON7>
- [13] Braun, J. (2010). On Kolmogorov's Superposition Theorem and its Applications: A Nonlinear Model for Numerical Function Reconstruction from Discrete Data Sets in Higher Dimensions. *Suedwestdeutscher Verlag fuer Hochschulschriften*. – 192 p.
- [14] Chart.js [Электронный ресурс]. – URL: <https://www.chartjs.org/docs/latest/>
- [15] React: the library for web and native user interfaces. [Электронный ресурс]. – URL: <https://react.dev>
- [16] Pyodide [Электронный ресурс]. – URL: <https://pyodide.org/en/stable/>
- [17] FastAPI [Электронный ресурс]. – URL: <https://fastapi.tiangolo.com>
- [18] Goodfellow, I., Bengio, Y., Courville, A. (2016). *Deep Learning*. – Cambridge, MIT Press. – 800 p.
- [19] Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
- [20] Семенов К.К., Тараскин А.С., Васин А.В. Программа обработки изображений, получаемых на устройствах электрофоретического разделения нуклеиновых кислот в агарозном геле, с интерактивной функцией разделения пересекающихся пиков на спектрограммах. Свидетельство о регистрации программы для ЭВМ RU 2023610644, 12.01.2023. Заявка № 2022685850 от 23.12.2022.
- [21] Семенов К.К., Тараскин А.С., Васин А.В. Программа обработки изображений, получаемых на устройствах электрофоретического разделения нуклеиновых кислот в агарозном геле, с автоматическим разделением пересекающихся пиков на спектрограммах. Свидетельство о регистрации программы для ЭВМ RU 2023610918, 16.01.2023. Заявка № 2022685876 от 23.12.2022.