

УДК: 519.6

Суррогатный нейросетевой метод восстановления поля течения из однородного поля итерациями в расчетах стационарных турбулентных течений

М. Н. Петров^а, С. В. Зими́на^б

Московский физико-технический институт,
Россия, 141707, г. Долгопрудный, Институтский пер., д. 9

E-mail: ^а mikhail.petrov@phystech.edu, ^б sofya.zimina@phystech.edu

*Получено 04.12.2024, после доработки — 07.02.2024.
Принято к публикации 10.03.2025.*

Последние годы получило широкое распространение применение нейросетевых моделей для решения задач аэродинамики. В основном такие модели, обученные по некоторому набору ранее полученных решений, позволяют предсказывать решения новых задач и являются в некотором смысле алгоритмами интерполяции. Альтернативным подходом может служить построение нейросетевого оператора, представляющего собой нейросетевую модель, которая воспроизводит поведение численного метода решения задачи. Такая модель позволяет находить решение задачи итерациями. В работе рассматривается вариант построения такого оператора с применением нейронной сети типа UNet с пространственным механизмом внимания для решения задач обтекания на прямоугольной равномерной сетке, общей для обтекаемого тела и поля течения. Для уточнения полученного решения предлагается и исследуется механизм коррекции решения. Анализируется вопрос устойчивости такого алгоритма решения стационарной задачи, проводится сравнение с некоторыми другими вариантами его построения: прием с продвижением вперед (pushforward trick), позиционное встраивание. Рассматривается вопрос выбора набора итераций для формирования обучающей выборки. Оценивается поведение решения при многократном применении нейросетевого оператора.

Демонстрация метода приводится для случая обтекания скругленной пластины турбулентным потоком воздуха с различными вариантами скругления при фиксированных параметрах набегающего потока с числом Рейнольдса $Re = 10^5$ и числом Маха $M = 0,15$. Поскольку течения с такими параметрами набегающего потока можно считать несжимаемыми, исследуются непосредственно только компоненты скорости. При этом нейросетевая модель, используемая для построения оператора, имеет общий декодер для обеих компонент скорости. Проводится сравнение полей течения и профилей скорости по нормали и по обводу тела, полученных нейросетевым оператором и численно. Анализ проводится как на пластине, так и на скруглении. Результаты моделирования подтверждают, что нейросетевой оператор позволяет находить решение с высокой точностью устойчивым образом.

Ключевые слова: аэродинамика, турбулентность, нейросетевой оператор, сверточная нейронная сеть, UNet, механизм внимания

Исследование выполнено за счет гранта Российского научного фонда № 23-71-01031 (<https://rscf.ru/project/23-71-01031/>).

UDC: 519.6

A surrogate neural network method for restoring the flow field from a homogeneous field by iterations in calculations of steady turbulent flows

M. N. Petrov^a, S. V. Zimina^b

Moscow Institute of Physics and Technology,
9 Institutskii lane, Dolgoprudny, 141707, Russia

E-mail: ^a mikhail.petrov@phystech.edu, ^b sofya.zimina@phystech.edu

*Received 04.12.2024, after completion — 07.02.2024.
Accepted for publication 10.03.2025.*

In recent years, the use of neural network models for solving aerodynamics problems has become widespread. These models, trained on a set of previously obtained solutions, predict solutions to new problems. They are, in essence, interpolation algorithms. An alternative approach is to construct a neural network operator. This is a neural network that reproduces a numerical method used to solve a problem. It allows to find the solution in iterations. The paper considers the construction of such an operator using the UNet neural network with a spatial attention mechanism. It solves flow problems on a rectangular uniform grid that is common to a streamlined body and flow field. A correction mechanism is proposed to clarify the obtained solution. The problem of the stability of such an algorithm for solving a stationary problem is analyzed, and a comparison is made with other variants of its construction, including pushforward trick and positional encoding. The issue of selecting a set of iterations for forming a train dataset is considered, and the behavior of the solution is assessed using repeated use of a neural network operator.

A demonstration of the method is provided for the case of flow around a rounded plate with a turbulent flow, with various options for rounding, for fixed parameters of the incoming flow, with Reynolds number $Re = 10^5$ and Mach number $M = 0.15$. Since flows with these parameters of the incoming flow can be considered incompressible, only velocity components are directly studied. At the same time, the neural network model used to construct the operator has a common decoder for both velocity components. Comparison of flow fields and velocity profiles along the normal and outline of the body, obtained using a neural network operator and numerical methods, is carried out. Analysis is performed both on the plate and rounding. Simulation results confirm that the neural network operator allows finding a solution with high accuracy and stability.

Keywords: aerodynamics, turbulence, neural operator, convolutional neural network, UNet, attention

Citation: *Computer Research and Modeling*, 2025, vol. 17, no. 2, pp. 179–197 (Russian).

The research was carried out at the expense of a grant from the Russian Science Foundation No. 23-71-01031 (<https://rscf.ru/project/23-71-01031/>).

Введение

Численное решение уравнений Навье – Стокса в задачах аэродинамики является непростой задачей и имеет долгую, богатую историю. С развитием машинного обучения и нейронных сетей последние годы набирают популярность суррогатные методы численного моделирования, позволяющие решать аэродинамические задачи как предсказание нейронной сети, обученной на наборе данных, которые чаще всего являются численными решениями некоторых других аэродинамических задач. Обычно база данных решений составляется для какого-то конкретного класса задач [Hanna et al., 2020; Morimoto et al., 2021; Зими́на, Петро́в, 2022; Petrov, Zimina, 2024; Chen, Thuerey, 2023; Buzzicotti, 2023]. На данный момент уже стали классическими подходы, которые позволяют по постановке задачи сразу предсказывать все ее решение. В этом смысле такие подходы можно рассматривать как способ интерполяции решения по уже полученным [Thuerey et al., 2020; Wandel, Weinmann, Klein, 2020; Zuo et al., 2022].

Альтернативным подходом может служить построение нейросетевого оператора [Hao et al., 2024; Herde et al., 2024]. Цель такого подхода — получать решение не как вариант интерполяции по данным с обучения, а итерациями, воспроизводя поведение численного метода, на данных решений которого этот оператор был обучен. Существенным преимуществом построения оператора потенциально могут быть лучшие возможности в нахождении решения даже для тех классов задач, которые не были представлены в обучающей выборке [Azizzadenesheli et al., 2024]. При этом с возникновением новых потенциальных возможностей у такого подхода возникают и фактические трудности, связанные с устойчивостью метода. В ходе решения задачи нейросетевым оператором итерациями могут накапливаться ошибки, которые чрезмерно искажают решение. Для решения этой проблемы исследователи предлагают использовать такие подходы как прием с продвижением вперед, прием с временным объединением [Brandstetter, Worrall, Welling, 2022], улучшение разрешения низкочастотной части спектра решения [Lippe et al., 2024], позиционное встраивание, пространственный механизм внимания [Rahman et al., 2024].

В работе проводится исследование принципиальной возможности построения нейросетевого оператора для решения задачи обтекания итерациями, когда поле течения и обтекаемое тело представлены на общей равномерной сетке. Ранее в статье [Петров, Зими́на, 2024] была построена суррогатная модель интерполяционного типа, позволяющая предсказывать решение по геометрии обтекаемого тела. Эта модель показала высокую точность результата. В частности, там решение получалось также на равномерной прямоугольной сетке и приводился способ восстановления поля течения в пристенной области и нахождения трения на стенке с использованием уравнений метода пристенной декомпозиции [Petrov et al., 2020; Petrov et al., 2022; Chikitkin et al., 2020]. Поэтому в этой статье будет сделан акцент именно на восстановлении самого поля течения, в предположении, что пристенная область может быть восстановлена тем же образом. В качестве базовой архитектуры нейронной сети будет использоваться та же, что и в статье [Петров, Зими́на, 2024]. Она уже включает в себя пространственный механизм внимания, что может быть полезно и при построении нейросетевого оператора для получения решения устойчивым образом [Rahman et al., 2024]. Для улучшения получаемого решения в работе будет предложен и рассмотрен механизм коррекции решения, который представляет собой нейронную сеть той же архитектуры. В этом случае решение будет получаться в два этапа: предсказание и коррекция. Вместе с этим для сравнения будут рассматриваться еще два способа, которые могли бы повысить точность получаемого решения: прием с продвижением вперед и позиционное встраивание. Валидация метода была проведена для задачи обтекания скругленной пластины с различными вариантами скругления. Тест показал, что нейросетевой оператор, построенный на основе архитектуры с пространственным механизмом внимания и последующей коррекцией результата, позволяет получать решение с высокой точностью. Оценка точности проводилась путем сравнения профилей скорости по нормали и по обводу тела.

Статья обладает следующей структурой. В разделе «Методы решения стационарных задач внешней аэродинамики» представлены общая идея построения нейросетевого оператора, способ получения датасета для его обучения и способ переинтерполяции решения. В разделе «Нейросетевой оператор» представлены способ описания геометрии решаемой задачи, архитектура используемой нейронной сети и способы устойчивого получения решения итерациями. Также рассматривается вопрос выбора итераций для обучения оператора. В разделе «Результаты» приводятся результаты применения нейросетевого оператора для получения решения в задаче обтекания скругленной пластины с различными вариантами скругления. Проводится анализ результатов в части получения профилей скорости потока для разных способов получения решения. Там же оценивается поведение решения при многократном применении нейросетевого оператора. В разделе «Заключение» подводятся итоги по полученным результатам, делаются выводы о возможных улучшениях предложенного метода.

Методы решения стационарных задач внешней аэродинамики

Математическая модель и метод получения численного решения

Для моделирования турбулентных течений в работе решаются уравнения Рейнольдса с помощью неявного конечно-объемного численного метода с замыканием по модели Спаларта–Аллмараса. Невязкие потоки получаются из решения задачи Римана о распаде разрыва, вязкие потоки и источниковые члены вычисляются с помощью аппроксимации пространственных производных. В результате система уравнений Рейнольдса сводится к системе нелинейных ОДУ. Применение неявной схемы Эйлера приводит к системе алгебраических уравнений. Данная система решается с использованием метода LU-SGS [Chikitkin et al., 2018]. Все численные решения получаются с помощью кода лаборатории «Flowmodellium» (МФТИ) [Петров и др., 2018].

Нейросетевой оператор для получения суррогатного решения

Для построения нейросетевого оператора A по базе расчетов, полученных численным методом с сохранением поля течения на каждой итерации, переинтерполированного на прямоугольную сетку, обучается модель типа UNet для восстановления поля течения из однородного поля итерациями. Восстановление производится авторегрессионным способом, то есть по решению, полученному на предыдущем шаге, получается решение на следующем. Поэтому на обучении минимизируется одношаговая функция потерь

$$L_{\text{one-step}} = L(A\mathbf{u}_k, \mathbf{u}_{k+1}), \quad (1)$$

где \mathbf{u}_k — численное решение на шаге k , \mathbf{u}_{k+1} — численное решение на шаге $k + 1$. Для получения гладкого решения применяется пространственный механизм внимания. Как и в численном методе, поле течения сначала инициализируется однородным образом, а далее итеративно уточняется. При этом на каждом шаге модели известны граничные условия, параметры набегающего потока и геометрия обтекаемого тела. Текущая итерация описывается текущим полем течения.

Переинтерполяция решения

Для обучения нейросетевого оператора сошедшееся решение, полученное численно, и решения, полученные на предыдущих итерациях, локализуются в области интереса у тела так, чтобы покрыть ту его часть, где течение меняется наиболее значительно. Затем эта область переинтерполируется на равномерную сетку с одинаковым шагом по пространственным направлениям. Шаг сетки при этом стоит выбирать из соображений разрешения требуемых физических масштабов. В частности, в исследовании шаг задается так, чтобы можно было разрешить пристенную область уравнениями метода пристенной декомпозиции. Получаемые таким образом

поля течения на равномерной сетке будут в дальнейшем использоваться для обучения нейросетевого оператора, а получаемая суррогатная модель не будет зависеть от исходной расчетной вычислительной сетки.

Рассмотрим неравномерную сетку, составленную из четырехугольников. Пусть индекс $i = 0$ будет указывать непосредственно на ячейку, в которой происходит интерполяция, индексы $i = 1, 4$ указывают на ячейки-соседи. Тогда для интерполяции воспользуемся формулой

$$u(x, y) = u_0 + (x - x_0)\alpha + (y - y_0)\beta.$$

Наклоны α и β могут быть выбраны так, чтобы сумма

$$\sum_{i=1}^4 [u_0 + (x_i - x_0)\alpha + (y_i - y_0)\beta - u_i]^2$$

достигала минимума. Здесь и ранее (x_i, y_i) — центры соответствующих ячеек. Минимум находится с помощью метода наименьших квадратов.

Поскольку сверточная нейронная сеть с двумерными свертками изначально разработана для работы с изображениями, которые имеют двумерную матрицу, а пиксели имеют одинаковый размер, то применение переинтерполяции поля течения на равномерную вычислительную сетку позволяет органично применить сверточную нейронную сеть для обучения суррогатной модели без необходимости адаптации к исходной вычислительной сетке, на которой были получены решения для обучения модели. Что в общем случае невозможно для неструктурированных вычислительных сеток.

Нейросетевой оператор

Нейросетевой оператор представляет собой сверточную нейросеть с двумерными свертками и архитектурой «энкодер – декодер» типа UNet [Ronneberger, Fischer, Brox, 2015], обученную на парах решений, взятых с предыдущей и следующей итераций численного метода. Причем на вход вместе с полем течения в модель передается также ее геометрия. В архитектуру сети добавлен пространственный механизм внимания для лучшего разрешения пространственной информации.

Способ описания геометрии тела

Для описания геометрии тела используются функция расстояния со знаком (SDF, signed distance function) и маскирование области течения. Обозначим тело как Ω , границу тела — как $\delta\Omega$. Определим расстояние от точки пространства до границы тела как

$$d(x, \delta\Omega) = \inf_{y \in \delta\Omega} d(x, y),$$

где \inf — точная нижняя грань значения функции расстояния между точками d . Тогда SDF определим как

$$\text{SDF}(x) = \begin{cases} -d(x, \delta\Omega), & x \in \Omega, \\ d(x, \delta\Omega), & x \notin \Omega. \end{cases} \quad (2)$$

Помимо того что SDF явно выделяет ту область, которая принадлежит телу (в ней $\text{SDF} < 0$), также SDF явно указывает, какая часть поля течения является пристеночной (в ней SDF принимает небольшие положительные значения), что вместе с использованием механизма внимания позволяет лучше учитывать физику течения в пристенной области.

Дополнительно с SDF производится простое маскирование области. Для этого ячейки, центры которых принадлежат телу, маскируются нулем, а ячейки, принадлежащие полю течения, —

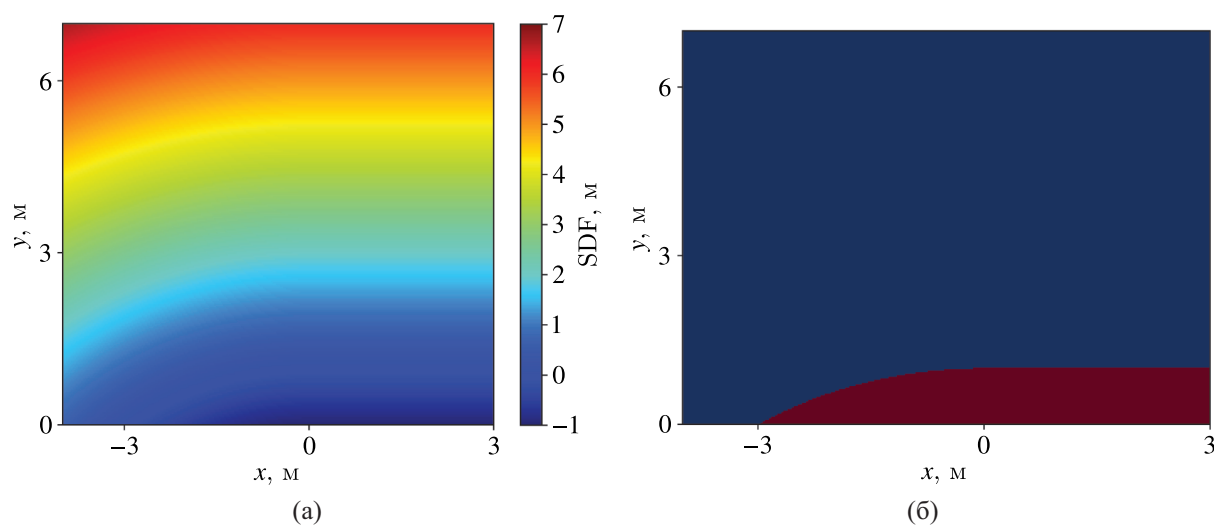


Рис. 1. Описания области течения с телом с помощью функции расстояния со знаком (SDF) (слева) и с помощью маскирования (справа) для задачи обтекания скругленной пластины

единицей. Ранее было замечено [Петров, Зими́на, 2024], что применение SDF вместе с маскированием позволяет нейросети лучше отделять поле течения от тела. Описания области течения с телом, полученные с помощью SDF и маскирования, составляют два канала входа нейронной сети (рис. 1).

Архитектура нейронной сети

Опишем архитектуру сверточной нейронной сети типа «энкодер – декодер» с двумерными свертками с общим декодером для обеих компонент скорости, лежащую в основе нейросетового оператора. Схематично архитектура представлена на рис. 2. На вход нейросети приходит описание области поля течения вместе с телом, полученное с помощью SDF (2) и маскирования, и само поле течения на выбранной итерации для двух компонент скорости, объединенные вместе в четыре канала. Выходом нейронной сети являются u_x и u_y компоненты скорости поля течения. Для обучения модели скорость течения в области тела полагается равной нулю.

Энкодер и декодер нейронной сети состоят из четырех блоков каждый. Один блок энкодера состоит последовательно из двух сверточных слоев, после каждого из которых стоят пакетная нормализация и функция активации (ReLU), дополненных слоем объединения (average pooling), используемым для уменьшения размеров карты признаков. Один блок декодера последовательно состоит из слоя повышения разрешения (upsampling), используемого для увеличения размеров карты признаков и двух сверточных слоев с пакетной нормализацией и активацией, как и в энкодере. Как и в классической архитектуре UNet, карты признаков сверточных слоев энкодера передаются в соответствующие слои декодера (как указано на рис. 2), но не напрямую, а через пространственный механизм внимания (attention gate), что позволяет лучше учитывать пространственную информацию, выделяя наиболее значимые области [Oktay et al., 2018].

Отметим некоторые детали реализации нейронной сети. Для уменьшения размеров карты признаков в энкодере использовался слой объединения, усредняющий значения в рамках заданного окна (average pooling). В декодере для увеличения размеров карты признаков использовался upsampling, который увеличивает размеры, восстанавливая значения увеличенной карты признаков с помощью интерполяции. Используемая интерполяция также влияет на гладкость решения. В нейросети для этого используется бикубическая интерполяция. Ранее в статье [Петров, Зими́на, 2024] исследовались варианты представленной архитектуры по способу задания

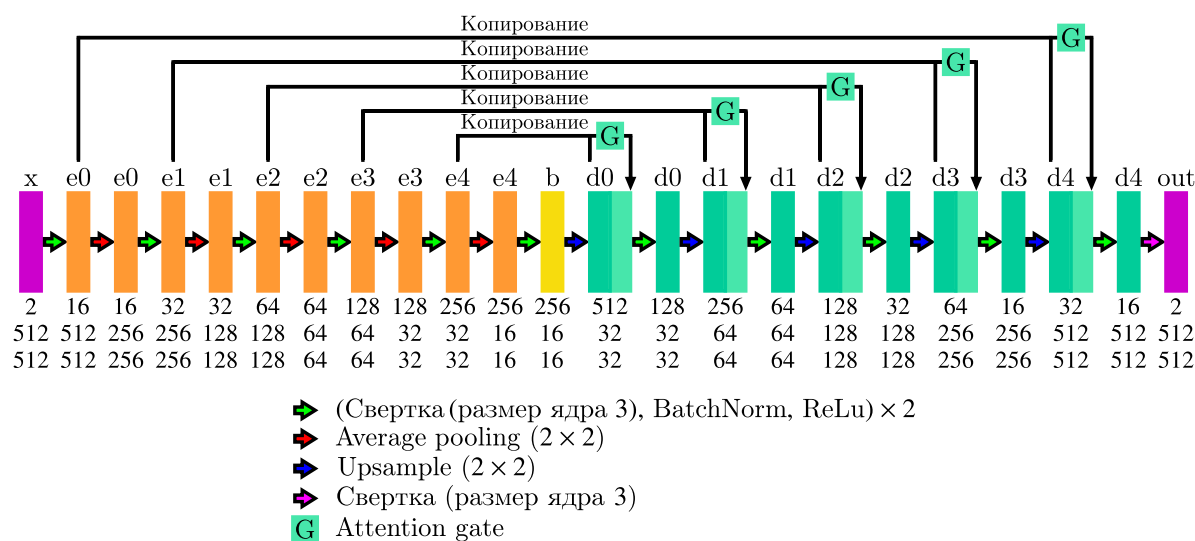


Рис. 2. Архитектура нейросети нейросетевого оператора. Average pooling — слой объединения, используемый для уменьшения размеров карты признаков. Upsample — слой повышения разрешения. Attention gate — пространственный механизм внимания. BatchNorm — пакетная нормализация. ReLU — функция активации

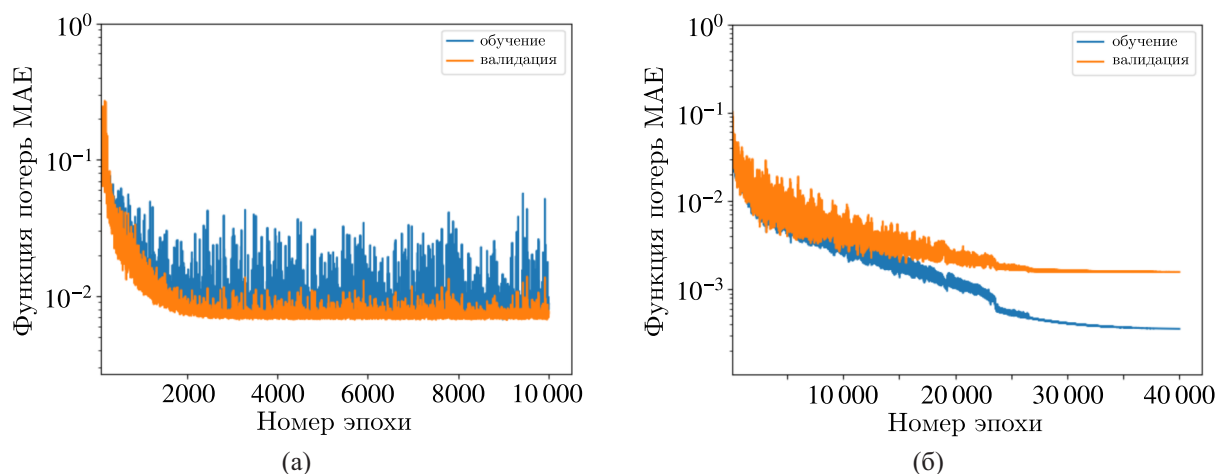


Рис. 3. График зависимости функции потерь от номера эпохи при обучении нейронной сети. Слева — обучение нейросетевого оператора, справа — обучение корректора

слоев и используемой интерполяции. Было показано, что предложенный здесь вариант наиболее оптимальный.

Для обучения нейронной сети в качестве функции потерь использовалась средняя абсолютная ошибка (MAE). Для минимизации функции потерь использовался алгоритм AdamW [Loshchilov, Hutter, 2017]. Для подбора темпа обучения (learning rate) использовался экспоненциальный планировщик (exponential scheduler). Обучение проводилось 10 000 эпох, что не является вычислительно затратным, если обучающая выборка невелика. График обучения на датасете, используемом в статье, представлен на рис. 3 (слева).

Прием с продвижением вперед

В статье [Brandstetter, Worrall, Welling, 2022] был предложен метод, позволяющий, по мнению авторов, сделать нейросетевой оператор более устойчивым — прием с продвижением вперед

(pushforward trick). Решение задачи итерациями при применении нейросетевого оператора может приводить к проблеме — неустойчивости, вызванной накоплением ошибки. Пусть \mathbf{u}_k — точное решение на шаге k . Тогда, при применении нейросетевого оператора к решению на шаге k , получится решение на шаге $k + 1$, отличающееся от точного на некоторую ошибку ϵ : $A\mathbf{u}_k = \mathbf{u}_{k+1} + \epsilon$. Чтобы оператор был менее чувствительным к ошибкам, возникающим на предыдущем шаге, необходимо обучить его предсказывать решение на следующем шаге с учетом ошибки, возникающей на предыдущем. Для этого предлагается к обычной авторегрессионной функции потерь (1), используемой при обучении оператора, добавить стабилизирующую функцию потерь

$$L_{\text{stab}} = L(A(\mathbf{u}_k + \epsilon), \mathbf{u}_{k+1}), \quad (3)$$

где возмущенное решение на шаге k было получено тем же нейросетевым оператором по точному решению с шага $k - 1$: $A\mathbf{u}_{k-1} = \mathbf{u}_k + \epsilon$.

Позиционное встраивание

Еще один способ, который может повлиять на устойчивость при решении задачи итерациями с помощью нейросетевого оператора, — это непосредственная передача информации о номере текущей итерации с помощью позиционного встраивания. Для этого в нейросетевую модель, представляющую используемый нейросетевой оператор и описанную ранее, после каждого блока из двух сверточных слоев с пакетной нормализацией и функцией активации добавляется позиционный слой. Позиционный слой состоит последовательно из функции активации (SiLU) и линейного слоя, размер входа которого соответствует размеру позиционного встраивания, а размер выхода — числу выходных каналов блока нейронной сети. Позиционная информация кодируется вектором. Вектор позиции прибавляется к смысловому вектору на входе модели, после чего результат подается в нейросеть. В качестве вектора позиции используются тригонометрические векторы [Vaswani, 2017]. В общем случае использование позиционного встраивания при построении нейросетевого оператора является одновременно как преимуществом, так и недостатком. С одной стороны, модель может лучше выучить как именно должен вести себя оператор на конкретной итерации. С другой стороны, модель привязывается к конкретным итерациям, и в случае, если набор используемых итераций будет другой, модель будет необходимо переучивать.

Коррекция результатов решения задачи итерациями

Для того чтобы улучшить качество решения, полученного нейросетевым оператором, в исследовании предлагается проводить коррекцию решения. Для этого после обучения оператора получают решения с помощью него итерациями с однородного поля течения для всех постановок из обучающей выборки. Затем из решений, полученных оператором, и сошедшихся решений, полученных численно, формируется новая обучающая выборка. По этой выборке обучается еще одна модель на основе той же архитектуры. Поскольку в используемой выборке существенно меньше данных, чем при обучении оператора, такая модель учится существенно быстрее. Так, если в обучающей выборке было M постановок задач, для каждой из которых было получено сошедшееся решение за N итераций, то размер обучающей выборки для обучения нейросетевого оператора — $M \cdot N$, а корректора — M . Обучение модели корректора проводилось 40 000 эпох (см. рис. 3 (справа)) с той же функцией потерь, оптимизатором и планировщиком, что и у основной модели нейросетевого оператора.

Вопрос выбора итераций для обучения. Формирование обучающей выборки

В исследовании расчет считался сошедшимся, если l_1 -норма невязки падала в 10^6 раз. В статье [Петров и др., 2018] было показано, что получившееся в таком случае решение при

турбулентном моделировании дает высокую точность результата, близкую к эксперименту. Для постановок задач из обучающей выборки расчет сходился примерно на итерации 100 000, а для обучения оператора брались итерации из той части расчета, где поле течения менялось существенно — невязка падала примерно в 1000 раз (итерации до 4000, поле значимо меняется не только в пристенной области).

В статье [Brandstetter, Worrall, Welling, 2022], на которую ссылаются и многие другие авторы, занимающиеся построением нейросетевых операторов, предлагается использовать прием с временным объединением (the temporal bundling trick) — предсказывать по предыдущему шагу K следующих шагов сразу совместно, обосновывая это лучшей устойчивостью оператора. По реализации такой подход, в сущности, аналогичен использованию на обучении не всех, а только некоторых определяющих итераций численного метода. Поэтому из 4000 итераций в исследовании будут использованы 8. Большое количество итераций, проводимых численным методом, обусловлено тем, что необходимо разрешать очень мелкую первую пристеночную ячейку. Для нейросетевого оператора после переинтерполяции решения на равномерную сетку для задач, рассматриваемых в статье, первая пристеночная ячейка больше примерно в 150–200 по сравнению с исходной вычислительной сеткой. Поэтому нейросетевая модель может различить не все изменения в поле течения, происходящие при численном решении, а только те, что происходят на более крупном масштабе (в согласии с отношением размеров первых пристеночных ячеек). Этим обусловлено количество итераций, используемых на обучении оператора. Также номера итераций выбирались экспоненциально, в согласии с характером убывания ошибки. В частности, в исследовании это были итерации с номерами 32, 46, 108, 251, 584, 1066, 4000, 100 000.

Результаты

Построение и исследование нейросетевого оператора проводились для решения задачи турбулентного обтекания скругленной пластины потоком воздуха с числом Рейнольдса $Re = 10^5$ и числом Маха $M = 0,15$ и варьированием формы скругления. Исходная расчетная область имеет размеры $(X, Y) = [-50, 50] \times [0, 50]$, где левая граница $x_{in} = -50$ — входная граница, правая граница $x_{out} = 50$ — выходная граница. На верхней границе и на нижней границе до скругления ставилось граничное условие «симметрия». Справа от точки $(x, y) = (0, 1)$ параллельно нижней границе расположена пластина, слева — скругление до нижней границы расчетной области. На рис. 4 представлены поле течения у тела и маска тела.

Для задания скругления пластины при определении геометрии окружность с центром в точке с координатой $x = 0$ под крайней точкой пластины (точка $(1, 0)$) строилась так, чтобы она выходила из этой точки до пересечения с осью ($y = 0$). Угол между двумя радиусами, проходящими через крайнюю точку пластины, и эту точку пересечения однозначно определяет радиус окружности и описывает вариант геометрии (см. рис. 5).

Как и в статье [Петров, Зимина, 2024], рассматривалось семь вариантов скругления. Пять вариантов с $\alpha \in [30, 45, 60, 75, 90]$ использовались для обучения нейросетевого оператора, два варианта с $\alpha \in [37, 68]$ — для валидации. Для всех вариантов были проведены турбулентные расчеты кодом «Flowmodellium» с использованием модели турбулентности Спаларта–Аллмараса. Все расчеты проводились на неструктурированной сетке со сгущением к скругленной пластине так, чтобы первый пристеночный шаг имел безразмерное расстояние $y^+ \approx 1$. Для всех расчетов из расчетной области выделялось поле течения в области $(X, Y) = [-11, 3] \times [0, 14]$ — та часть области у скругления, где течение меняется наиболее значительно. Поле течения в выделенной области переинтерполировалось на равномерную сетку, состоящую из 512×512 квадратных ячеек со стороной $h \approx 0,0273$, что для рассматриваемого течения соответствует $y^+ \approx 150$ –200. Для

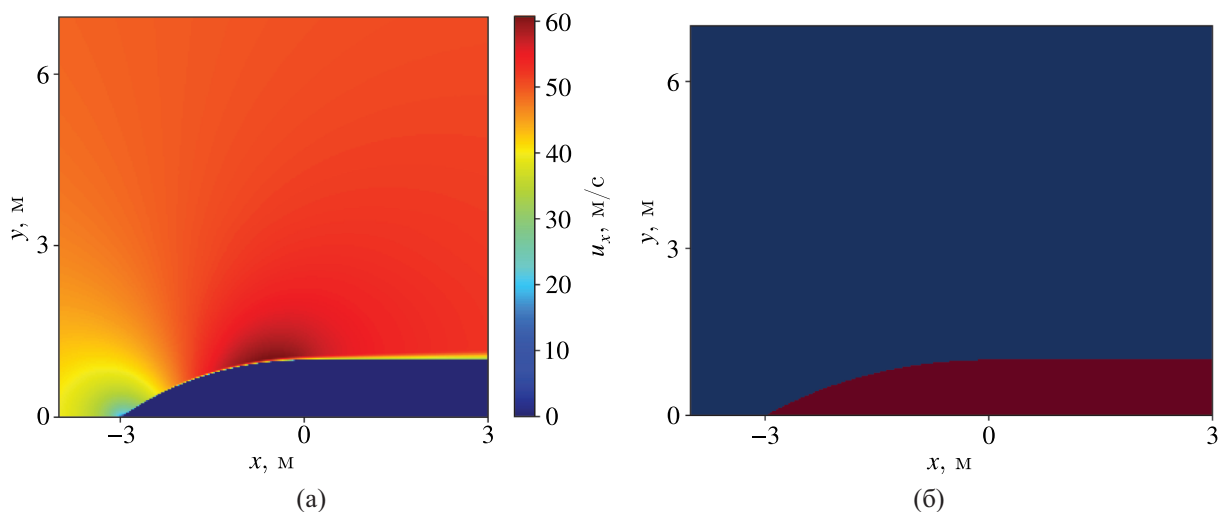


Рис. 4. Проинтерполированное поле течения (м/с) для компоненты скорости u_x (слева) и маска тела (справа) для задачи обтекания скругленной пластины (красным обозначено тело, синим — поле течения)

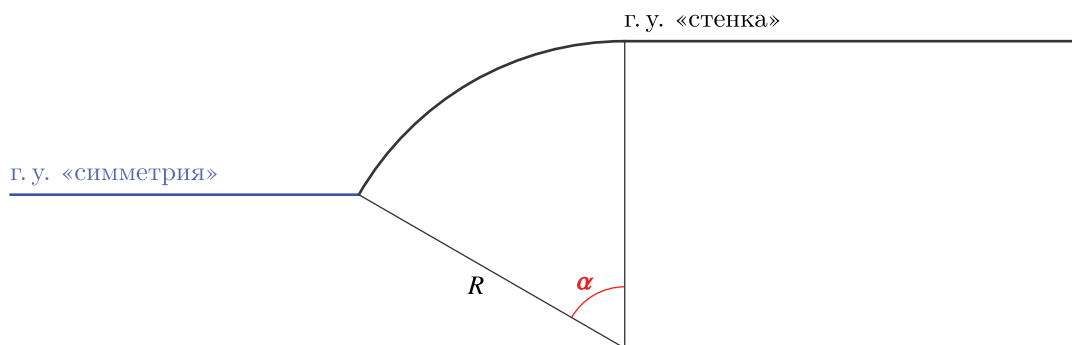


Рис. 5. Определение варианта скругления при задании геометрии. R — радиус скругления, α — угол между двумя радиусами, проходящими через точку $(1, 0)$ и точку пересечения окружности с осью $y = 0$

обучения нейросетевого оператора компоненты поля скорости пересчитывались по следующей формуле с помощью функции «симметричный логарифм»:

$$f(x) = \text{sign}(x) \cdot \ln(1 + \text{sign}(x) \cdot x).$$

В формуле $\text{sign}(x) = \frac{x}{|x|}$ для $x \neq 0$ и $\text{sign}(0) = 0$. Это преобразование упрощает обучение, так как придает больше значимости течению у стенки, где скорости ближе к нулю. При анализе результатов выход нейросети обратным преобразованием пересчитывался, чтобы получить исходные значения скорости в м/с. Рассмотрим результаты обучения нейросетевого оператора на валидационных тестах. Сравним сначала поля течения для обеих компонент скорости для случая $\alpha = 37$ и $\alpha = 68$ без учета и с учетом коррекции.

Из рис. 6 и 7 можно видеть, что нейросетевой оператор, как без коррекции, так и с коррекцией, итерациями позволяет получать решение, близкое к решению, полученному численным методом. Тем не менее базовый вариант решения (без коррекции), полученный нейросетевым оператором, имеет существенное отличие от численного как раз в пристенной области. Такое поведение решения объясняется ошибкой метода. Поскольку в пристенной области скорость в целом ближе к нулю, ошибки, вносимые оператором, сказываются на точности там значительно, чем в целом по полю. Коррекция поля течения существенно снижает эту ошибку для обеих компонент скорости.

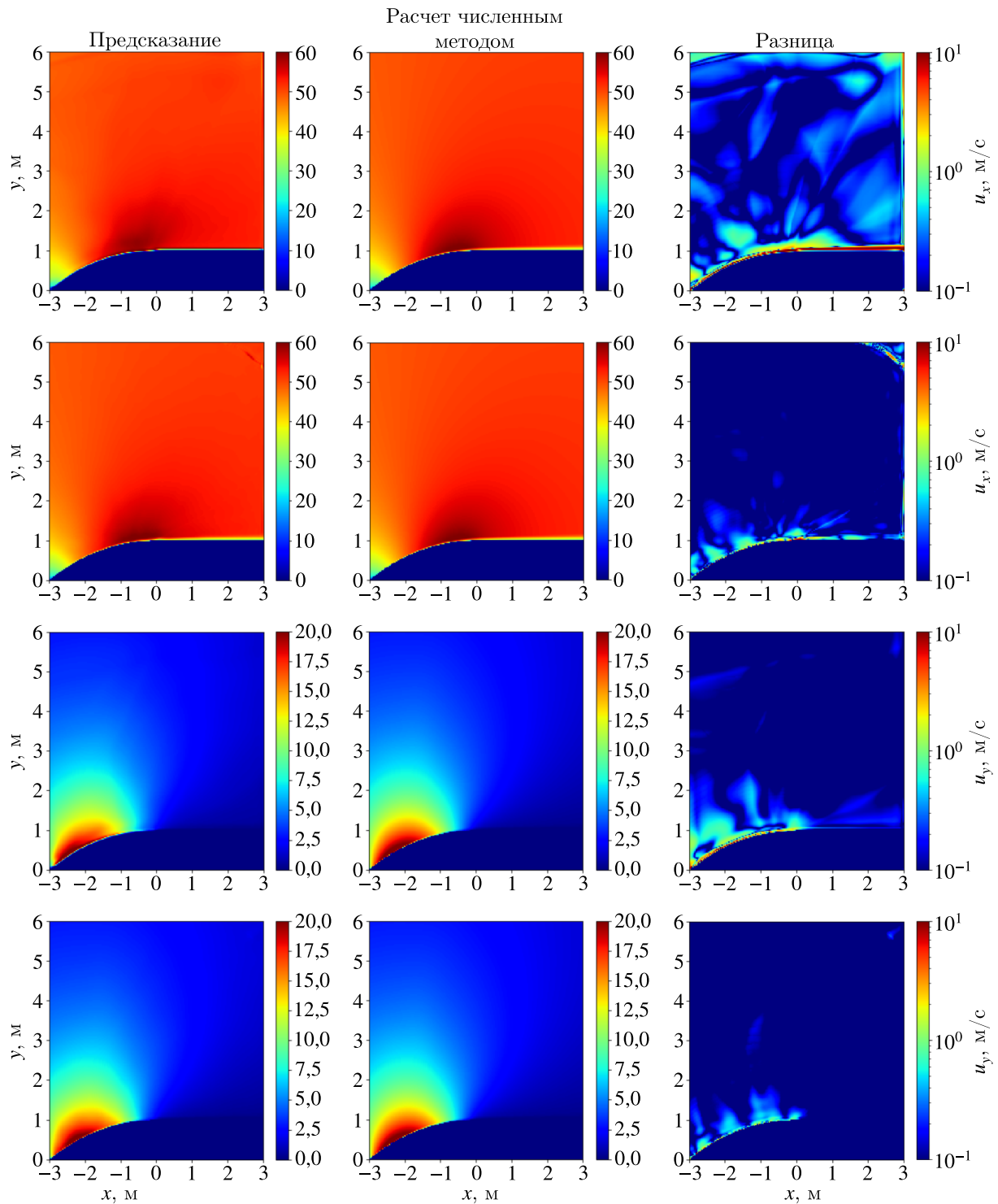


Рис. 6. Поля течения для геометрии с $\alpha = 37^\circ$. Первый столбец — полученное нейросетевым оператором, второй — численным методом, третий — разница между ними. Первая строка — поле для компоненты скорости u_x без коррекции результата, вторая — с коррекцией. Третья строка — поле для компоненты скорости u_y без коррекции результата, четвертая — с коррекцией

Теперь сравним профили скоростей по нормали к стенке. Сначала сравним профили скоростей на пластине у ее края ($x = 0$) и на некотором удалении от края ($x = 2$). На рис. 8

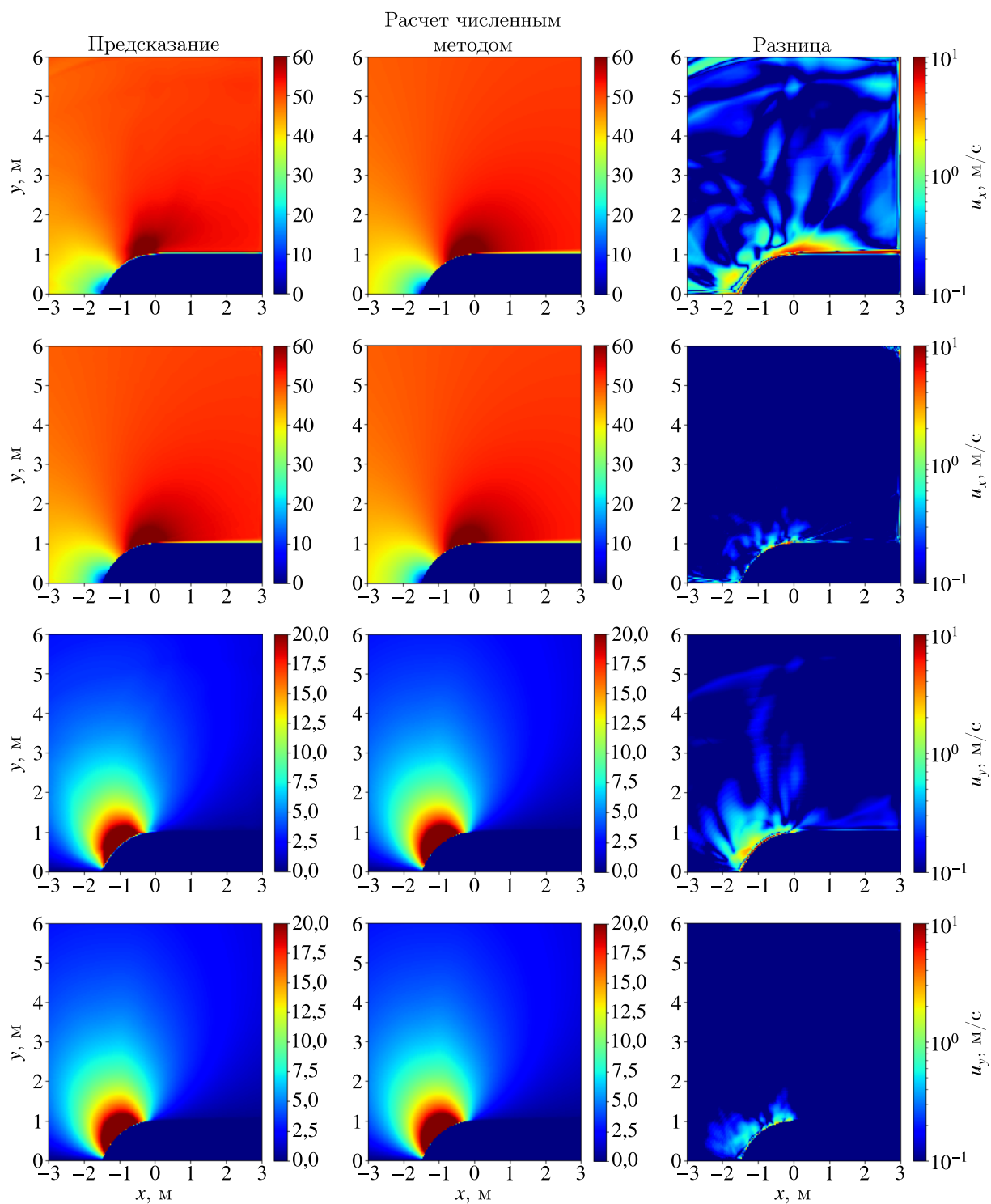


Рис. 7. Поля течения для геометрии с $\alpha = 68^\circ$. Первый столбец — полученное нейросетевым оператором, второй — численным методом, третий — разница между ними. Первая строка — поле для компоненты скорости u_x без коррекции результата, вторая — с коррекцией. Третья строка — поле для компоненты скорости u_y без коррекции результата, четвертая — с коррекцией

представлены профили, полученные численно, и еще по четыре профиля (для двух вариантов скругления и обеих компонент скорости), полученных разными вариантами построения нейросе-

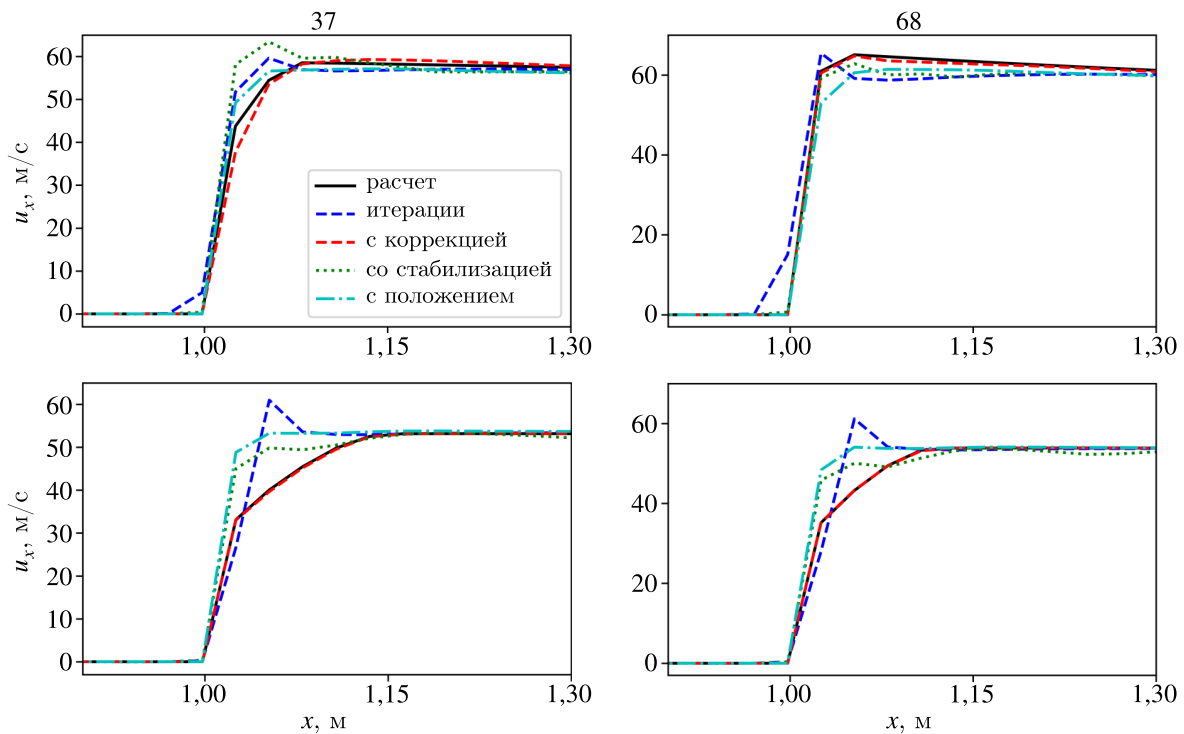


Рис. 8. Профили компоненты скорости u_x (м/с) на пластине по нормали к стенке у ее края ($x = 0$) (верхний ряд) и на удалении от края ($x = 2$) (нижний ряд). Левый столбец — для геометрии с $\alpha = 37^\circ$, правый — с $\alpha = 68^\circ$. Расчет — значения, полученные численно, итерации — полученные нейросетевым оператором (базовый вариант), с коррекцией — вариант итераций с коррекцией результата, со стабилизацией — вариант итераций с использованием стабилизирующей функции потерь на обучении, с положением — вариант итераций с использованием позиционного встраивания

тевого оператора: итерации — профиль, полученный нейросетевым оператором, с коррекцией — вариант итераций с коррекцией результата, со стабилизацией — вариант итераций с использованием стабилизирующей функции потерь на обучении, с положением — вариант итераций с использованием позиционного встраивания. Значения скоростей для построения графиков берутся из центров ячеек прямоугольной сетки. Из рисунка видно, что у края пластины рядом со скруглением нейросетевой оператор немного размывает границу тела. Все остальные варианты построения оператора решают эту проблему. Наибольшее совпадение профилей получается в варианте с коррекцией результата. Можно видеть почти точное совпадение графика варианта с коррекцией с графиком решения, полученного численно (расчет). Варианты со стабилизацией и с положением позволяют убрать выбросы в решении (острые пики у графика «итерации»), но остаются все еще недостаточно точными непосредственно у стенки.

Рассмотрим профили скорости по нормали к стенке на скруглении для геометрии с $\alpha = 37^\circ$ (от точки на скруглении с $x = -2$) и для геометрии с $\alpha = 68^\circ$ (от точки на скруглении с $x = -1$). Положение нормалей выбиралось до исследования. Так как в этом случае нормаль не проходит непосредственно через центры ячеек сетки, значения скоростей профиля интерполируются по прямоугольной сетке. Как и для профилей на пластине, на рис. 9 представлены профили для всех рассматриваемых вариантов построения оператора вместе с профилем, полученным численно (расчет). В целом видно, что все варианты, кроме варианта с положением, достаточно близки к профилю, получаемому численно. Наилучшее совпадение, как и ранее, у варианта с коррекцией. Хотя выбросы у стенки для базового варианта здесь не так выражены, как для пластины, тем не менее вариант со стабилизацией немного сглаживает решение.

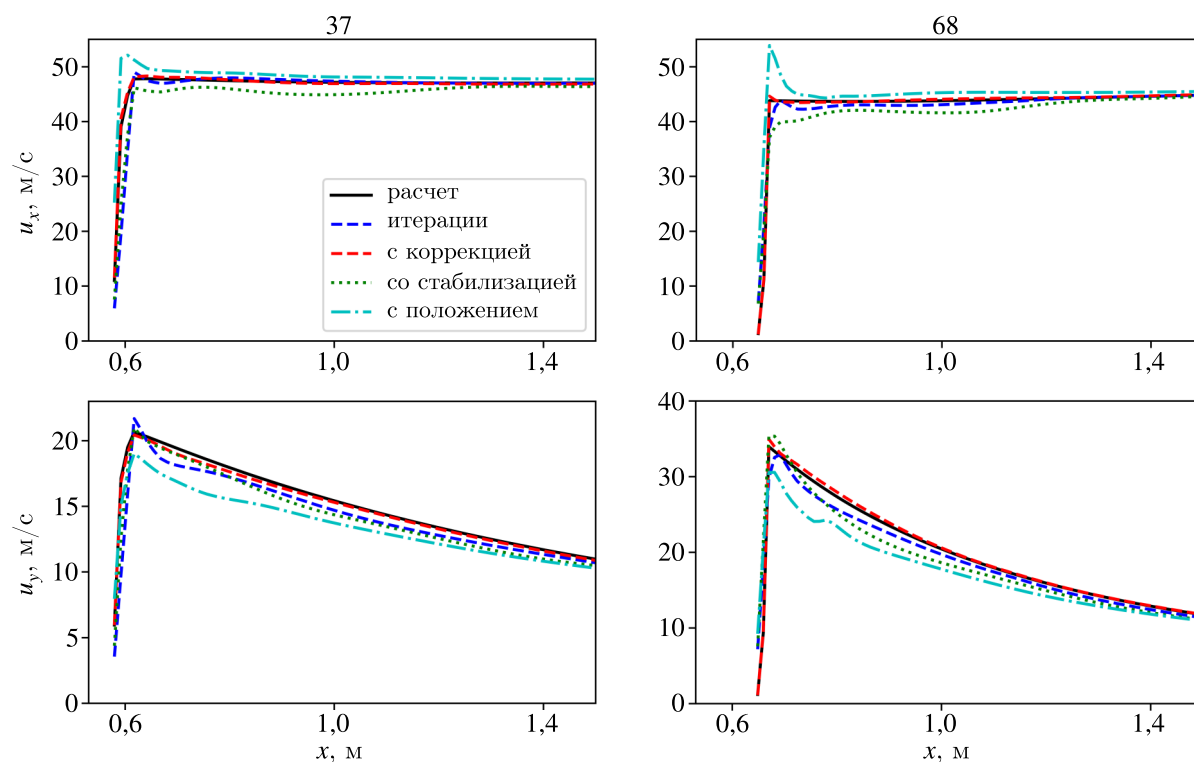


Рис. 9. Профили компонент скорости (м/с) на скруглении по нормали к стенке для геометрии с $\alpha = 37^\circ$ ($x = -2$, левый столбец) и для геометрии с $\alpha = 68^\circ$ ($x = -1$, правый столбец). Расчет — значения, полученные численно, итерации — полученные нейросетевым оператором, с коррекцией — вариант итераций с коррекцией результата, со стабилизацией — вариант итераций с использованием стабилизирующей функции потерь на обучении, с положением — вариант итераций с использованием позиционного встраивания

Теперь рассмотрим графики скоростей по обводу тела у пластины и у скругления. Фактически это наиболее важные графики исследования, так как именно от того, какова скорость у стенки, зависит расчет таких характеристик, как, например, трение на стенке. Ранее в статье [Петров, Зими́на, 2024] было показано, как, зная скорость у стенки, можно рассчитать трение с хорошей точностью, даже на такой грубой сетке, как в рассматриваемых тестах. Для пластины скорости возьмем непосредственно в центре первой пристеночной ячейки. Поскольку граница ячеек прямоугольной сетки не обязана по построению совпадать со стенкой, в данном случае получилось, что для пластины центр первой пристеночной ячейки находится на расстоянии $y \approx 0,0254$, что соответствует $y^+ \approx 180\text{--}200$. Из рис. 10 видно, что на пластине наибольшее совпадение с расчетом только у варианта с коррекцией.

По обводу у скругления скорости будем получать из интерполяции по полю течения. Возьмем расстояние от стенки $y = 0,022$, что соответствует $y^+ \approx 140\text{--}180$. Это расстояние задается непосредственно от скругления реальной геометрии, а не от маски тела. Это важно, поскольку маска тела является приближением реальной геометрии тела на прямоугольной сетке и имеет на скруглении отличающуюся границу. Поэтому задание расстояния от маски, а не от фактической геометрии, геометрическое описание которой известно, может исказить результат моделирования. Также среди всех точек по обводу будем брать только те, которые интерполируются только по полю течения. Это требование связано с тем, что для интерполяции значения в точке используется билинейная интерполяция, которая вычисляется по четырем значениям в соседних узлах. Так как значения скоростей на маске полагаются равными нулю, использование узлов, попадающих на маску, будет искажать результат интерполяции. Из рис. 11 видно, что, хотя вариант

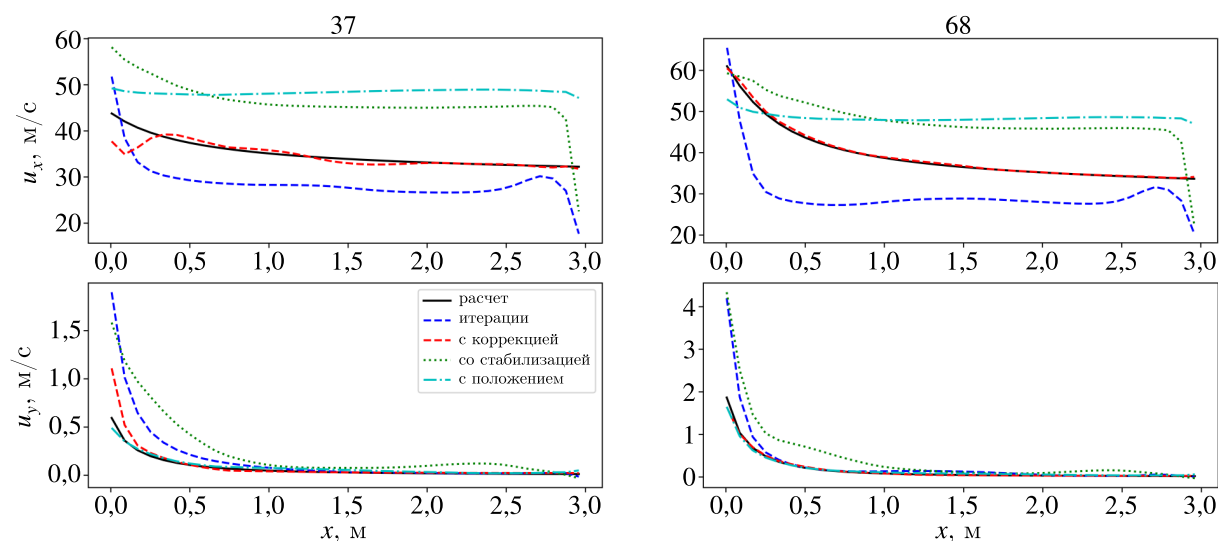


Рис. 10. Профили компонент скорости по обводу у пластины. Левый столбец — для геометрии с $\alpha = 37^\circ$, правый — с $\alpha = 68^\circ$. Расчет — значения, полученные численно, итерации — полученные нейросетевым оператором, с коррекцией — вариант итераций с коррекцией результата, со стабилизацией — вариант итераций с использованием стабилизирующей функции потерь на обучении, с положением — вариант итераций с использованием позиционного встраивания

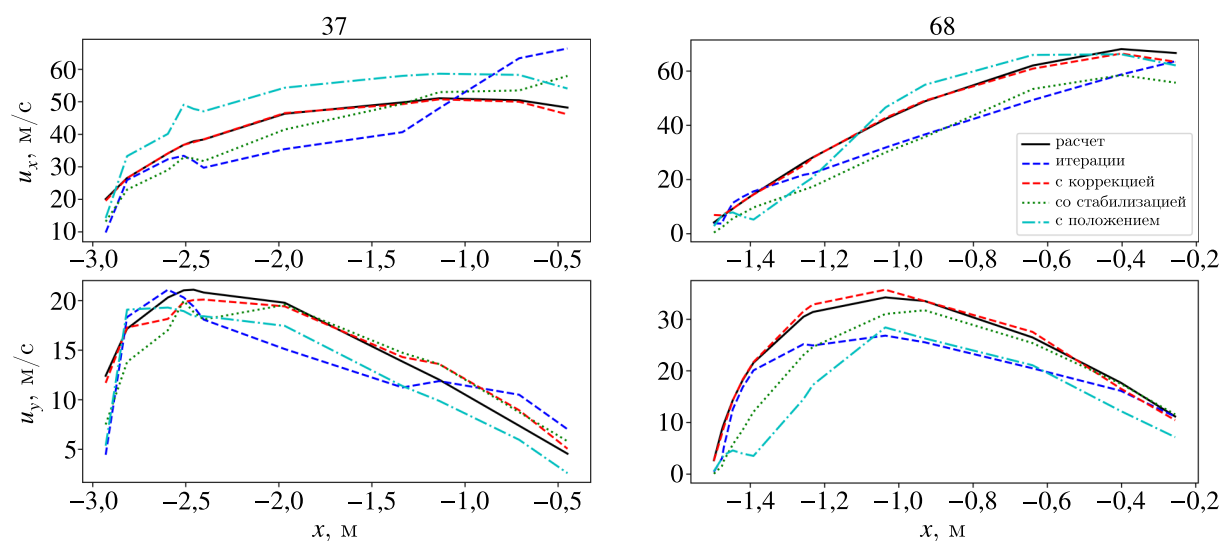


Рис. 11. Профили компонент скорости по обводу у скругления. Левый столбец — для геометрии с $\alpha = 37^\circ$, правый — с $\alpha = 68^\circ$. Расчет — значения, полученные численно, итерации — полученные нейросетевым оператором, с коррекцией — вариант итераций с коррекцией результата, со стабилизацией — вариант итераций с использованием стабилизирующей функции потерь на обучении, с положением — вариант итераций с использованием позиционного встраивания

с коррекцией снова показывает наилучшую точность, вариант со стабилизацией демонстрирует тоже относительно хорошее качество и обладает лучшей гладкостью, чем базовые вариант и вариант с положением.

На основании проведенных тестов можно сделать вывод, что полученные результаты демонстрируют принципиальную возможность построения нейросетевого оператора, способного получать решения устойчивым образом на равномерной прямоугольной сетке вместе с обтекае-

мым телом. Наиболее перспективным при этом выглядит подход с коррекцией решения. Также в отдельных случаях хорошо себя показывает и вариант со стабилизацией. В дальнейшем кажется целесообразным объединить эти два подхода в один для достижения лучшей точности получаемого решения.

Анализ поведения решения при многократном применении нейросетевого оператора

Для вариантов задач, рассматриваемых в статье, до сходимости всегда использовалось восемь итераций. Предполагалось, что, применяя нейросетевой оператор к равномерному полю, расчет должен сойтись к восьмой итерации. В этом смысле особый интерес вызывает вопрос о том, что будет с решением, если продолжить далее применять оператор к решению уже после восьмой итерации. Такая проверка в каком-то смысле является аналогом исследования численного метода на сходимость. Рассмотрим профили компонент скорости по обводу у пластины и на скруглении с увеличением числа итераций. На рис. 12 и 13 представлены профили для 8, 16, 32, 64 итераций. Из графиков видно, что если решение, полученное нейросетевым оператором, оказывается достаточно близким к численному, то последующие итерации не оказывают существенного влияния на решение. Такое поведение решения особо видно на скруглении (см. рис. 13). Это можно трактовать как сходимость решения для численного метода. И наоборот, заметные отклонения от численного решения, полученные через 8 итераций, в дальнейшем будут нарастать, что видно на рис. 12, и особенно для варианта геометрии с $\alpha = 37^\circ$. Тем не менее в целом даже увеличение общего количества итераций нейросетевого оператора в 16 раз не приводит к существенному искажению решению, что подтверждает корректность рассматриваемого метода.

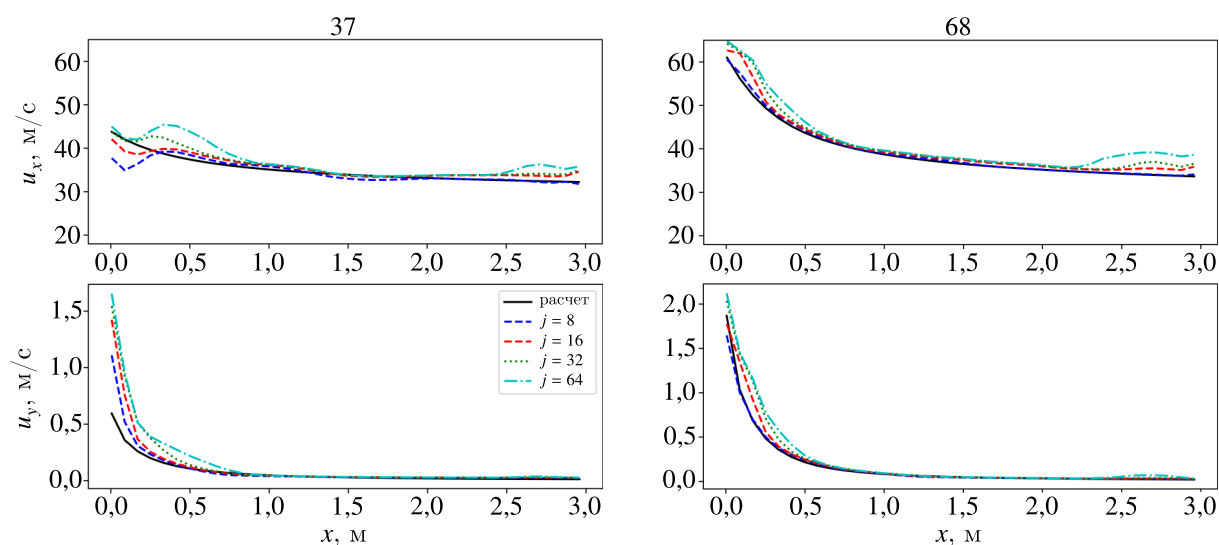


Рис. 12. Профили компонент скорости по обводу у пластины для разного количества итераций, совершенных нейросетевым оператором с коррекцией. Левый столбец — для геометрии с $\alpha = 37^\circ$, правый — с $\alpha = 68^\circ$. Расчет — значения, полученные численно, $j = 8, 16, 32, 64$ — полученные оператором, число указывает на количество итераций

Заключение

В работе был рассмотрен вопрос построения нейросетевого оператора для решения стационарных задач внешней аэродинамики итерациями. Метод был протестирован на модельной

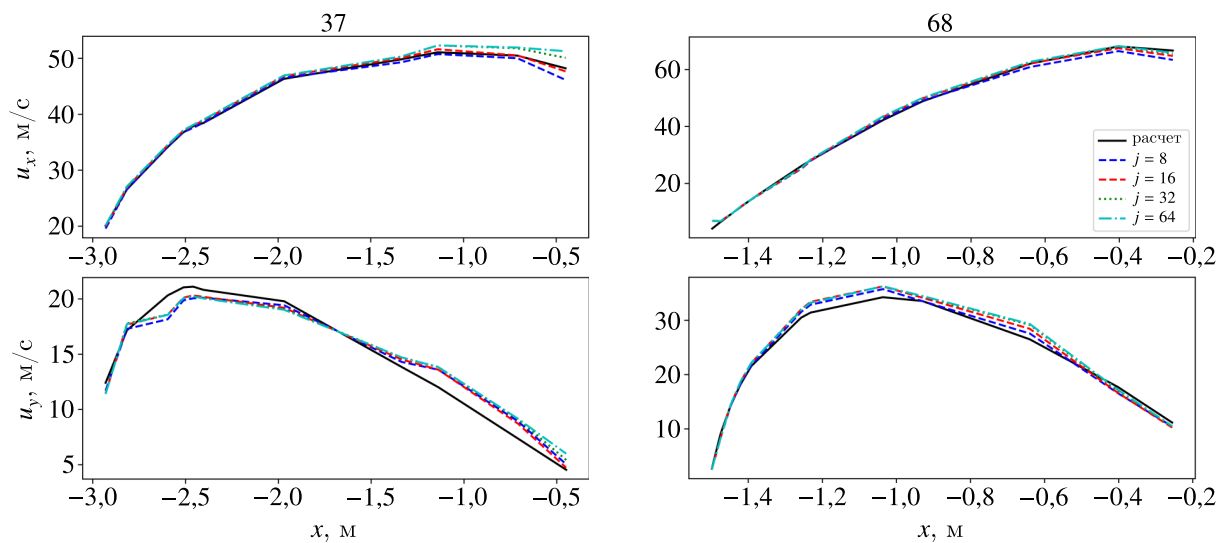


Рис. 13. Профили компонент скорости по обводу у скругления для разного количества итераций, совершенных нейросетевым оператором с коррекцией. Левый столбец — для геометрии с $\alpha = 37^\circ$, правый — с $\alpha = 68^\circ$. Расчет — значения, полученные численно, $j = 8, 16, 32, 64$ — полученные оператором, число указывает на количество итераций

задаче обтекания скругленной пластины дозвуковым турбулентным потоком воздуха с различными вариантами скругления. Основная идея метода заключалась в том, что нейросетевой оператор, обученный на базе расчетов, полученной некоторым численным методом, сможет предсказывать именно поведение самого численного метода при пересчете решения от итерации к итерации. Потенциально такой подход может быть более робастным для решения задач, на которых сам оператор обучен не был. В этой же работе была исследована сама принципиальная возможность построения нейросетевого итерационного метода решения задач, рассматривался вопрос устойчивого получения решения в подходе, когда поле течения рассматривается вместе с телом на общей прямоугольной сетке. Было установлено, что построение такого оператора принципиально возможно. Основным вызовом для метода является получение устойчивого решения. Рассматривалось несколько способов, которые могли бы позволить получать решение устойчивым образом: прием с продвижением вперед, позиционное встраивание, коррекция результатов решения. Наиболее перспективным для подхода, рассматриваемого в статье, выглядит способ с коррекцией решения. Также был рассмотрен вопрос выбора количества итераций на обучении оператора. Исследовано поведение решения с увеличением количества итераций, проводимых обученным оператором. Было показано, что рост количества итераций, проводимых уже обученным оператором, не ведет к существенному искажению решения.

Для достижения большей точности планируется провести улучшение разрешения низкочастотной части спектра решения. Также в будущем планируется проверить, как поведет себя обученный нейросетевой оператор на задачах, не входящих в класс задач из обучающей выборки.

Список литературы (References)

- Зими́на С. В., Петро́в М. Н. Уточнение поля течения с помощью сверточных нейронных сетей в задачах внешней аэродинамики // Успехи кибернетики. — 2022. — Т. 3, № 1. — С. 44–48. — <https://doi.org/10.51790/2712-9942-2022-3-1-6>
- Zimina S. V., Petrov M. N. Utochnenie polya techeniya s pomoshch'yu svertochnykh neironnykh setei v zadachakh vneshnei aerodinamiki [Application of convolutional neural networks to flow fields refining in external aerodynamics problems] // Russian Journal of Cybernetics. — 2022. — Vol. 3, No. 1. — P. 44–48 (in Russian).

- Петров М. Н., Зими́на С. В.* Суррогатная нейросетевая модель для восстановления поля течения в серийных расчетах стационарных турбулентных течений с разрешением пристенной области // Компьютерные исследования и моделирование. — 2024. — Т. 16, № 5. — С. 1195–1216.
Petrov M. N., Zimina S. V. Surrogatnaya neurosetevaya model' dlya vosstanovleniya polya techeniya v seriinykh raschetakh statsionarnykh turbulentnykh techenii s razresheniem pristennoi oblasti [A surrogate neural network model for resolving the flow field in serial calculations of steady turbulent flows with a resolution of the nearwall region] // Computer research and modeling. — 2024. — Vol. 16, No. 5. — P. 1195–1216 (in Russian).
- Петров М. Н., Тамбова А. А., Титарев В. А., Утюжников С. В., Чикиткин А. В.* Программный комплекс Flow Modelling для расчета высокоскоростных течений сжимаемого газа // Журнал вычислительной математики и математической физики. — 2018. — Т. 58, № 11. — С. 1932–1954.
Petrov M. N., Tambova A. A., Titarev V. A., Utyuzhnikov S. V., Chikitkin A. V. FlowModelling software package for calculating high-speed flows of compressible fluid // Computational Mathematics and Mathematical Physics. — 2018. — Vol. 58, No. 11. — P. 1865–1886. — DOI: 10.1134/S0965542518110118 (Original Russian paper: *Petrov M. N., Tambova A. A., Titarev V. A., Utyuzhnikov S. V., Chikitkin A. V.* Programmnyi kompleks Flow Modelling dlya rascheta vysokoskorostnykh techenii szhimaemogo gaza // Zhurnal vychislitel'noi matematiki i matematicheskoi fiziki. — 2018. — Vol. 58, No. 11. — P. 1932–1954.)
- Azizzadenesheli K., Kovachki N., Li Z., Liu-Schiaffini M., Kossaiji J., Anandkumar A.* Neural operators for accelerating scientific simulations and design // Nature Reviews Physics. — 2024. — P. 1–9.
- Brandstetter J., Worrall D., Welling M.* Message passing neural PDE solvers // arXiv preprint. — 2022. — arXiv:2202.03376
- Buzzicotti M.* Data reconstruction for complex flows using AI: Recent progress, obstacles, and perspectives // Europhysics Letters. — 2023.
- Chen L. W., Thuerey N.* Towards high-accuracy deep learning inference of compressible flows over aerofoils // Computers & Fluids. — 2023. — Vol. 250. — 105707.
- Chikitkin A., Utyuzhnikov S., Petrov M., Titarev V.* Non-overlapping domain decomposition for modeling essentially unsteady near-wall turbulent flows // Computers & Fluids. — 2020. — 104506.
- Chikitkin A. V., Petrov M. N., Titarev V. A., Utyuzhnikov S. V.* Parallel versions of implicit LU-SGS method // Lobachevskii Journal of Mathematics. — 2018. — Vol. 39, No. 4. — P. 503–512. — DOI: 10.1134/S1995080218040054
- Hanna B. N., Dinh N. T., Youngblood R. W., Bolotnov I. A.* Machine-learning based error prediction approach for coarse-grid Computational Fluid Dynamics (CG-CFD) // Progress in Nuclear Energy. — 2020. — Vol. 118. — 103140. — DOI: 10.1016/j.pnucene.2019.103140
- Hao Z., Su C., Liu S., Berner J., Ying C., Su H., Zhu J.* Dpot: Auto-regressive denoising operator transformer for large-scale PDE pre-training // arXiv preprint. — 2024. — arXiv:2403.03542
- Herde M., Raonić B., Rohner T., Käppeli R., Molinaro R., de Bézenac E., Mishra S.* Poseidon: efficient foundation models for PDEs // arXiv preprint. — 2024. — arXiv:2405.19101
- Kashefi A., Mukerji T.* Physics-informed PointNet: A deep learning solver for steady-state incompressible flows and thermal fields on multiple sets of irregular geometries // Journal of Computational Physics. — 2022. — Vol. 468. — 111510.
- Lippe P., Veeling B., Perdikaris P., Turner R., Brandstetter J.* PDE-refiner: Achieving accurate long rollouts with neural PDE solvers // Advances in Neural Information Processing Systems. — 2024. — P. 36.
- Loshchilov I., Hutter F.* Decoupled weight decay regularization // arXiv preprint. — 2017. — arXiv:1711.05101
- Morimoto M., Fukami K., Zhang K., Nair A. G., Fukagata K.* Convolutional neural networks for fluid flow analysis: toward effective metamodeling and low dimensionalization // Theoretical and Computational Fluid Dynamics. — 2021. — Vol. 35, No. 5. — P. 633–658.
- Oktay O., Schlemper J., Folgoc L. L., Lee M., Heinrich M., Misawa K., Rueckert D.* Attention u-net: Learning where to look for the pancreas // arXiv preprint. — 2018. — arXiv:1804.03999

- Petrov M., Utyuzhnikov S., Chikitkin A., Smirnova N.* Extension of near-wall domain decomposition to modeling flows with laminar-turbulent transition // *Communications in Computational Physics*. — 2022. — Vol. 31, No. 2. — P. 645–668. — DOI: 10.4208/cicp.OA-2021-0123
- Petrov M., Utyuzhnikov S., Chikitkin A., Titarev V.* On extension of near-wall domain decomposition to turbulent compressible flows // *Computers & Fluids*. — 2020. — Vol. 202. — 104629.
- Petrov M., Zimina S.* An application of space-filling curves to improve results of turbulent aerodynamics modeling with convolutional neural networks // *Chinese Journal of Aeronautics*. — 2024. — Vol. 37, No. 2. — P. 81–92.
- Rahman M. A., George R. J., Elleithy M., Leibovici D., Li Z., Bonev B., Anandkumar A.* Pretraining codomain attention neural operators for solving multiphysics PDEs // *arXiv preprint*. — 2024. — arXiv:2403.12553
- Ronneberger O., Fischer P., Brox T.* U-net: Convolutional networks for biomedical image segmentation // *Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, proceedings*. — 2015. — P. 234–241.
- Thuerey N., Weissenow K., Prantl L., Hu X.* Deep learning methods for Reynolds-averaged Navier–Stokes simulations of airfoil flows // *AIAA Journal*. — 2020. — Vol. 58, No. 1. — P. 25–36.
- Vaswani A.* Attention is all you need // *Advances in Neural Information Processing Systems*. — 2017.
- Wandel N., Weinmann M., Klein R.* Learning incompressible fluid dynamics from scratch—towards fast, differentiable fluid models that generalize // *arXiv preprint*. — 2020. — arXiv:2006.08762
- Zuo K., Bu S., Zhang W., Hu J., Ye Z., Yuan X.* Fast sparse flow field prediction around airfoils via multi-head perceptron based deep learning architecture // *Aerospace Science and Technology*. — 2022. — Vol. 130. — 107942.