

УДК 004.94

## МОДЕЛИРОВАНИЕ ПРОЦЕССА ГОРЕНИЯ ВОДОРОДА ПРИ РАЗЛИЧНЫХ ДАВЛЕНИЯХ С ПОМОЩЬЮ НЕЙРОННОЙ СЕТИ

М. Ю. Мальсагов<sup>1</sup>, Е. В. Михальченко<sup>1</sup>, Я. М. Карандашев<sup>1,2</sup>,  
В. Ф. Никитин<sup>1</sup>

<sup>1</sup>Научно-исследовательский институт системных исследований РАН (ФГУ ФНЦ НИИСИ РАН)  
117218 Москва, malsagov@niisi.ras.ru, mikhalkhenkolena@yandex.ru, karandashev@niisi.ras.ru,  
vfnikitin77@rambler.ru

<sup>2</sup>Российский университет дружбы народов, 117198 Москва

Исследуется возможность решения задач химической кинетики с использованием искусственных нейронных сетей. Основная трудоемкость решения задач химической кинетики заключается в решении жесткой системы уравнений баланса, в правой части которой стоит интенсивность производства массы компонентов. Эта задача может быть выделена в отдельный этап решения системы обыкновенных дифференциальных уравнений внутри общего шага по времени глобальной задачи, и в данной работе рассматривается именно этот этап. Разработана достаточно простая модель, способная решить эту задачу, благодаря которой удалось добиться трехкратного ускорения вычислений по сравнению с численными методами. Полученная нейронная сеть работает в рекурсивном режиме и может предсказывать поведение химической многокомпонентной динамической системы на много шагов вперед.

Ключевые слова: численное моделирование химических процессов, горение, детонация, нейронные сети, глубокое обучение.

DOI 10.15372/FGV20230204

### ВВЕДЕНИЕ

Во многих случаях задача расчета газодинамических процессов включает в себя физико-химические взаимодействия. Расчет этих взаимодействий без сильных дополнительных упрощений, снижающих предсказательную способность математического моделирования, представляет собой сложную математическую задачу, традиционно решаемую методами, которые при всей возможной их оптимизации занимают большую, если не подавляющую часть расчетного времени. Если рассчитывается нестационарный процесс, в который входят такие физические явления, как детонация, переход к детонации, зажигание горючей смеси, распространение волн горения или детонации по системе, то получение данных даже за

весьма малое физическое время требует долгой работы суперкомпьютерных систем. Попытка же ускорить время расчета обычными для этого простыми методами, такими как огрубление расчетной сетки или экстраполяция результатов, полученных на начальном этапе расчета, на значительно больший временной период, не гарантирует отсутствия грубых отклонений численной модели от исходной математической. Тем самым, узким местом математического моделирования сложных газодинамических процессов становится расчет химических взаимодействий.

В последнее время с развитием графических ускорителей и методов глубокого обучения участились попытки аппроксимации химической кинетики искусственными нейронными сетями. Преимущества такого подхода очевидны: для нейронной сети не критично количество задействованных в процессе веществ (чем больше, тем лучше); подавляющее количество вычислений — это скалярное умножение матрицы на вектор, что можно эффективно распараллелить, т. е. получить значительное ускорение. При этом одна и та же архитектура сети может использоваться для различных кинети-

Работа выполнена в рамках государственного задания Федеральному научному центру «Научно-исследовательский институт системных исследований Российской академии наук» по темам FNEF-2022-0003 и FNEF-2022-0021.

Доложено на II Международной научной конференции «Физика и химия горения и процессов в экстремальных условиях», 12–16 июля 2022, Самара.

© Мальсагов М. Ю., Михальченко Е. В.,  
Карандашев Я. М., Никитин В. Ф., 2023.

ческих механизмов, достаточно изменить входной и выходной векторы, обучить сеть на новых данных.

Преимущества нейросетевого подхода продемонстрированы в недавних работах [1–5]. Например, в [1, 2] предлагают обучать по одной маленькой сети для каждого вещества, задействованного в процессе. Затем параллельно объединить эти сети в одну с общим входом и выходом. Хотя авторы не упоминают, с какими численными методами они сравнивают сеть, однако указывают 35-кратное ускорение. В работе [3] несколько одинаковых небольших сетей с остаточными связями (ResNet) соединяются последовательно, по количеству временных шагов, на которые делается аппроксимация. Такой подход позволяет достичь хорошей точности при прогнозировании на десятки, сотни шагов от стартового состояния.

Тем не менее, решения эти были неполны. Хотя в работах [1–3] и многих других давление в системе учитывалось явно, обучающие данные создавались для одного значения давления или небольшого диапазона. То есть, чтобы учесть большой интервал начальных условий, требуется обучать отдельную нейронную сеть для каждого небольшого диапазона или значения давления.

В данной работе нам удалось решить проблему с учетом давления, используя лишь одну небольшую нейронную сеть.

## ПОСТАНОВКА ЗАДАЧИ

Рассматривается задача горения водородно-воздушной смеси. Начальное состояние системы задается давлением, температурой и молярными плотностями водорода, кислорода и других соединений, образующихся в процессе горения, включая возможные радикалы. В физико-химические процессы вовлечены соединения  $H_2$ ,  $O_2$ ,  $H_2O$ ,  $OH$ ,  $HO_2$ ,  $H_2O_2$ ,  $H$ ,  $O$ , нейтральными остаются  $N_2$ ,  $Ar$ . В процессе горения меняется температура среды.

Для расчета выбран механизм GRI-Mech 3.0 (1999) [6]. Несмотря на то, что этот механизм разрабатывался прежде всего для расчета горения углеводородов, а также образования оксидов азота в процессе горения, он имеет полностью автономную подсистему реакций, описывающих горение кислородно-водородной смеси с фиксацией азота нейтральным. Технология, разрабатываемая в настоящей работе, легко переносится на любой другой подобный ме-

ханизм, в том числе узкоспециализированный. Целью же работы является именно выработка технологии решения кинетики нейронной сетью.

В данной подсистеме механизма GRI-Mech процесс преобразования веществ описывается 28 химическими реакциями. Азот и аргон не образуют химических соединений с другими веществами, а только влияют на скорость протекания реакций. Нами рассматривается в качестве химико-кинетического этапа изохорический процесс, в котором фиксируются плотность и сумма внутренней и химической энергии смеси. Поэтому молярная плотность нейтральных компонентов с течением времени никак не меняется, в отличие от остальных веществ. Таким образом, система состоит из 10 переменных (давление, температура и кислородно-водородные соединения) и двух компонентов, чьи значения не меняются, но учитывать эти значения все же нужно.

Математическая постановка задачи заключается в следующем. Решается система дифференциальных уравнений по числу компонентов  $N_C$  и одного алгебраического уравнения вида

$$\frac{d\chi_k}{dt} = \dot{\omega}_k(\chi_1, \dots, \chi_{N_C}; T), \quad (1)$$

$$E = \sum_{k=1}^{N_C} E_k(T)\chi_k = E_0.$$

Здесь  $\chi_k$  — молярная плотность компонента,  $\dot{\omega}_k$  — интенсивность производства одного моля компонента в единице объема,  $E$  — сумма внутренней и кинетической энергии единицы объема смеси,  $E_k$  — заданная функция зависимости внутренней энергии одного моля компонента от температуры.

В нашей задаче всего 10 компонентов, т. е. 11 уравнений в системе (1). Сохранение общей массы, а также массы каждого элемента в химических реакциях является следствием особой структуры интенсивностей их производства. Эти функции весьма сложны, что и делает этап расчета химической кинетики трудоемким.

Задача (1) решается как задача Коши для заданных начальных значений молярных плотностей и температуры. Практически вместо начальных молярных плотностей и температуры удобно задавать начальные значения молярных долей, температуры и давления.

## НАБОР ДАННЫХ

Для обучения сети необходим большой набор данных, поэтому потребовалось численное решение системы (1). Численная реализация основана на методе Новикова для жестких систем обыкновенных дифференциальных уравнений [7]. В заданных диапазонах генерировались начальные состояния топливной смеси: давление  $p_0 \in [0.1; 100]$  бар, температура  $T_0 \in [1\,000; 2\,000]$  К, ненормированные молярные доли компонентов  $\{C_{\text{H}_2}, C_{\text{O}_2}\} \in [0.1; 100]$  и  $\{C_{\text{N}_2}, C_{\text{Ar}}\} \in [0.1; 50]$ . Эти данные использовались для вычисления начальных молярных плотностей  $k$ -х веществ:

$$\chi_{k0} = \frac{p_0 C_{k0}}{RT_0 \sum_k C_{k0}} \quad [\text{моль/м}^3]. \quad (2)$$

Затем численным методом определялись молярные плотности веществ через равные интервалы времени  $dt$ . Эти таблицы состояний системы во времени использовались в дальнейшем для обучения и тестирования нейронной сети.

Важно отметить, что выбор правильного временного шага имеет первостепенное значение. В большинстве случаев процесс горения водородно-кислородной смеси протекает очень быстро. Поэтому, выбрав слишком большой или маленький шаг, можно получить просто постоянный процесс без каких-либо изменений. То есть получатся бессмысленные данные.

Большое влияние на скорость реакции оказывает давление. Чем больше давление, тем быстрее начинается и заканчивается горение. Также, согласно формуле (2), давление прямо влияет на молярную плотность вещества.

Чтобы как-то минимизировать ошибки выбора временного шага, мы решили сделать его функцией давления, так чтобы выполнялось соотношение

$$P \cdot dt \sim 0.1 \text{ [мкс]}. \quad (3)$$

Давление здесь в начальный момент времени является безразмерным и нормированным:

$$P = p_0[\text{Па}]/10^5[\text{Па}].$$

Состояние системы в любой момент времени определяется 12 параметрами: давлением  $P$ , температурой  $T$  и молярными плотностями десяти веществ. Всё это компоненты одного 12-мерного вектора, который должен служить входом и выходом нейронной сети. При

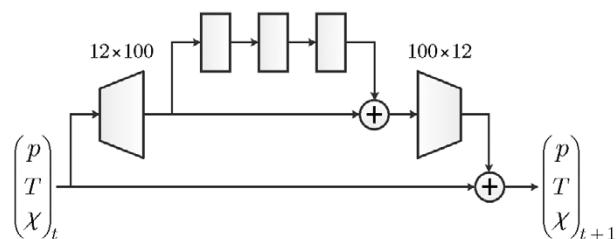


Рис. 1. Архитектура нейронной сети

этом каждый компонент этого вектора не только подчиняется своему закону во времени, но и меняется каждый в своем диапазоне значений: тысячи кельвин для температуры, десятки единиц для давления, а молярные плотности могут достигать как десятков и сотен единиц, так и величин порядка  $10^{-20}$ .

Поэтому ко всем данным применялось стандартное масштабирование:

$$\tilde{X} = \frac{\ln(\varepsilon + X) - M_X}{\sigma_X},$$

$$X = \left\{ P, T, \frac{\chi}{P_0} \right\}. \quad (4)$$

Здесь  $M_X$  и  $\sigma_X$  — среднее значение параметра и стандартное отклонение (покомпонентные), вычисленные на тренировочном наборе после логарифмирования, а  $\varepsilon = 10^{-20}$  — коэффициент, чтобы избежать нулевых значений молярных плотностей в начальный момент времени.

## АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ

На рис. 1 представлена архитектура нейронной сети. Сеть получилась пятислойной. Первый и последний слой переводят соответственно входной 12-мерный вектор в 100-мерное пространство и обратно. Средние три полносвязных слоя имеют по 100 нейронов с активационной функцией LeakyReLU. Присутствуют также прямые связи, передающие входной вектор без изменений на выход, как в остаточных нейронных сетях (ResNet). Сеть получилась простая, компактная, без каких-либо ухищрений. Единственное, стоит отметить, что молярные плотности азота и аргона остаются постоянными, поэтому они просто передаются на выход сети.

Такая сеть может предсказывать только на один шаг по времени. То есть на вход сети

подается начальная конфигурация  $X_t$ , после чего сеть предсказывает измененные значения через один шаг времени  $X_{t+1}$ , что сравнивается с желаемым значением  $\hat{X}_{t+1}$ , полученным численно. При этом сеть учится минимизировать разницу между  $X_{t+1}$  и  $\hat{X}_{t+1}$ . С другой стороны, в процессе функционирования потребуется предсказание на несколько шагов вперед, в идеале на несколько сотен или даже тысяч шагов. Такое предсказание можно получить, если обученную сеть запускать в рекуррентном режиме, т. е. выход сети подавать на вход и т. д. При этом очень важно, чтобы ошибка предсказания на один шаг была бы как можно меньше, поскольку при многократном запуске сети ошибка быстро накапливается и растет как минимум линейно.

В настоящей работе мы включили предсказание на несколько шагов в рекуррентном режиме в сам процесс обучения. Обучение сети проводилось сразу на несколько шагов, т. е. выход сети подавался на вход некоторое количество раз  $n_{step}$ , а функция потерь вычислялась по накопленным данным:

$$\text{Loss} = \sum_{t=1}^{n_{step}} \text{MSE}(Y_t, Y_t^{pred}), \quad (5)$$

где MSE — среднеквадратичная ошибка,  $Y_t$  и  $Y_t^{pred}$  — соответственно эталонный и предсказанный сетью выходы.

Благодаря суммированию ошибок предсказания на несколько шагов вперед, сеть автоматически учится учитывать рекуррентное накопление ошибки и минимизировать его. В экспериментах мы использовали  $n_{step} = 50$  шагов.

## РЕЗУЛЬТАТЫ

Для обучения и тестирования нейронной сети было создано три набора данных. Первый набор из 10 000 экспериментов предназначен для обучения. Обученная нейронная сеть тестировалась на базе из 5 000 экспериментов. Еще 4 000 экспериментов использовались для контроля процесса обучения. Каждый эксперимент содержит 500 временных шагов. Величина временного шага в каждом случае определялась из соотношения (3) по начальному давлению.

На рис. 1 представлена общая архитектура сети. У нее много параметров, которые можно менять, чтобы добиться лучшей производительности. Например, это количество нейронов в слоях, активационные функции и их внутренние параметры, алгоритм оптимизации и т. д. Мы провели множество экспериментов, варьируя параметры сети. За эталонную сеть примем архитектуру, показанную на рис. 1. В ней пять слоев по 100 нейронов в каждом. Средние три слоя с прямой связью будем называть блоком. Таких блоков может быть много вложено друг в друга, вместо среднего слоя вставляется целый блок и т. д., получается так называемая UNet-архитектура [8]. Активационная функция используется LeakyReLU с коэффициентом  $\text{slope} = 0.15$ . Алгоритм оптимизации при обучении — Adam. Вариации нейронных сетей, позволяющие представить, как изменение одного из параметров влияет на качество нейронной сети, приведены в таблице. В первой ее строке исходная сеть с параметрами, описанными ранее. Затем идет трехслойная сеть, в которой средние три слоя заменены одним. Работает такая сеть очень плохо, т. е. уменьшать количество слоев нельзя. Замена активационной функции на более простую (ReLU) также ухудшает ситуацию. Строки 4–6 показывают, что правильный подбор параметра активационной функции позволяет в два раза улучшить результат.

Дальнейшие эксперименты проводились уже при оптимальном параметре функции активации. Сеть с тремя вложенными блоками, очевидно, дала лучший результат, однако это сопровождается резким увеличением вычислительных затрат. Для нас минимальность вычисления является критичным требованием, поэтому этот подход придется отложить. Изменение алгоритма оптимизации, а также числа нейронов в слоях в обе стороны не дает заметного улучшения.

В итоге можно заключить, что исходная сеть с подправленным параметром активационной функции является оптимальной и дает наилучший результат (строка 5 в таблице).

На рис. 2 показана среднеквадратичная ошибка на каждом тестовом примере. Для удобства результат отсортирован, примеров в тестовом наборе было 5 000. Худшее значение составило  $0.037495792$ , а лучшее —  $5.92598 \cdot 10^{-7}$ . Конечно, эти числа мало что говорят о качестве работы нейронной сети. На рис. 3

Вариации нейронной сети

№ п/п	Нейронная сеть	Среднеквадратичная ошибка	Стандартное отклонение
1	Исходная сеть LeakyReLU (slope = 0.15)	0.000614116	0.005362478
2	3 центральных слоя заменены одним	0.015666055	0.049845003
3	ReLU	0.001828609	0.038570476
4	LeakyReLU (slope = 0.05)	0.00048006	0.002815006
5	LeakyReLU (slope = 0.01)	0.000306201	0.00167874
6	LeakyReLU (slope = 0.005)	0.000511011	0.002898796
7	3 блока LeakyReLU (slope = 0.01)	0.000198631	0.001040813
8	Adamax LeakyReLU (slope = 0.01)	0.000477337	0.002537064
9	50 нейронов в слое	0.003537873	0.017569872
10	150 нейронов в слое	0.000418604	0.00314917

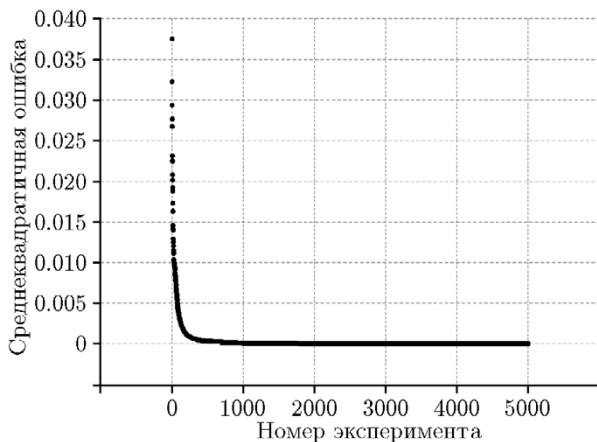


Рис. 2. Среднеквадратичная ошибка на каждом тестовом примере

представлен худший результат из тестовой выборки. Видно, что на поздних шагах, после 300-го, нейронная сеть начинает опережать истинные значения. При этом характер зависимости предсказывается точно. Отметим, что в этом конкретном примере процесс происходит с задержкой, т. е. большую часть времени ситуация спокойна, и только под конец начинается возгорание. Если посмотреть на остальные примеры, то можно заметить, что чем раньше начинается горение, тем лучше сеть описывает процесс, а при таких «запаздывающих» начинается опережать. Нейронная сеть смогла

обучиться необходимым зависимостям, однако она не способна должным образом описывать постоянные или сильно отложенные процессы. Вводя зависимость временного шага от давления, мы пытались автоматизировать подбор параметров, чтобы наблюдения всегда приходились на процесс возгорания. Тем не менее, очевидно, всех случаев нам учесть не удалось, и при медленных процессах сеть дает сбой на длительных дистанциях. Возможно, увеличение количества временных шагов в каждом примере в обучающей выборке решит эту проблему.

Сравнить нейронную сеть по количеству операций с численным методом невозможно. В отличие от сети, количество операций в использовавшемся численном методе меняется (например, шаг интегрирования подбирается в процессе и т. д.). Также современные фреймворки и графические ускорители позволяют параллельно запускать одну нейронную сеть на сотнях и тысячах входных векторов. Например, тестовая выборка из 5 000 примеров с 500 шагами считается менее чем за секунду на видеокарте NVIDIA GeForce RTX 2070 Super. Однако, реализовав сеть на языке Си, сравнив реальное время работы исполняемых файлов на одном процессоре, мы получили, что нейронная сеть работает в 2-3 раза быстрее без каких-либо оптимизаций и параллелизма.

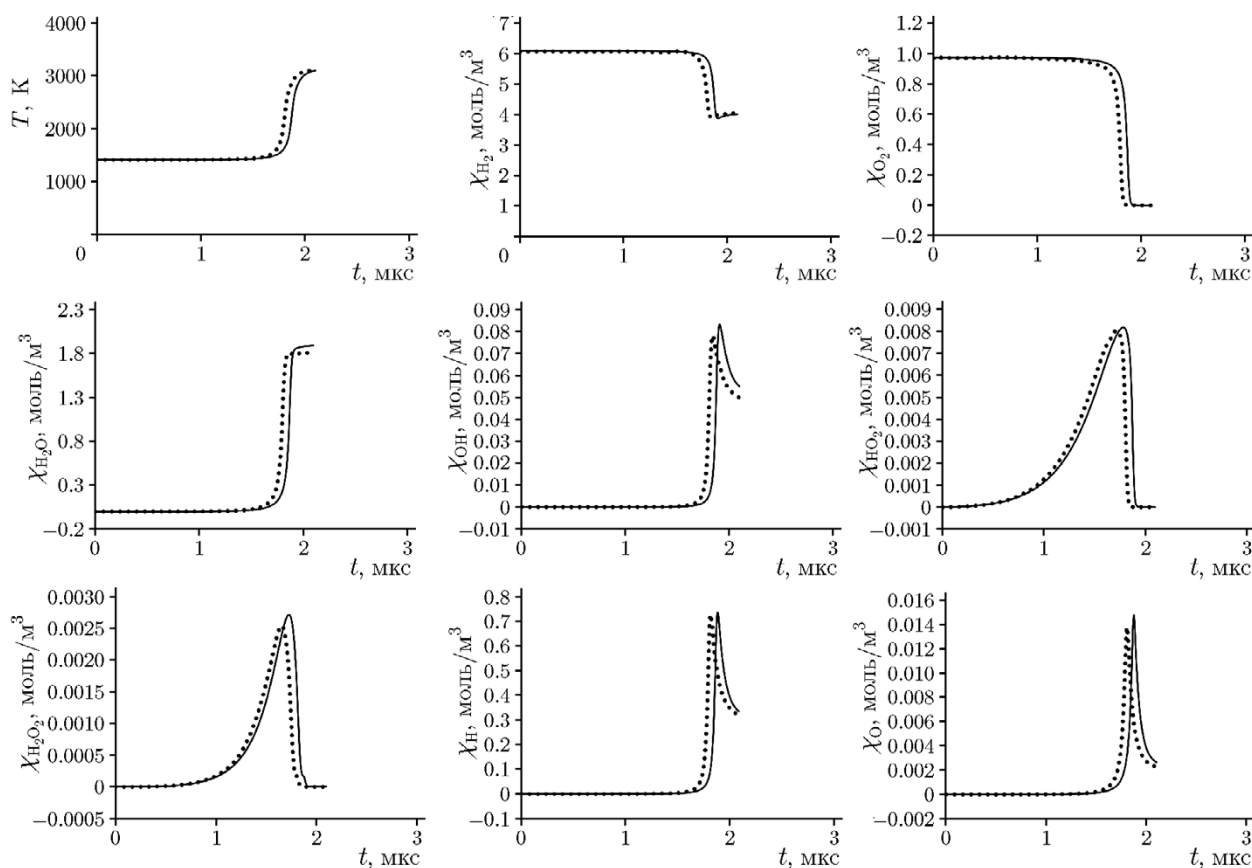


Рис. 3. Худший результат в тестовой выборке: сравнение нейронной сети (точки) и численного метода (сплошная кривая):

временной шаг  $dt = 0.0042$  мкс, 500 шагов

## ЗАКЛЮЧЕНИЕ

При компьютерном моделировании процессов горения наиболее вычислительно трудоемкая часть состоит в моделировании химической кинетики. Предложенная в данной работе нейронная сеть справляется с этой задачей и является быстрой, легко распараллеливаемой альтернативой трудоемкому численному решению при наличии достаточной обучающей выборки.

В данной работе нам удалось ввести в модель учет давления системы. Наличие давления в модели привело к тому, что не только данные значительно менялись по величине, но и сильно изменялась скорость происходящих процессов. Эти изменения делали невозможным обучение одной сети для всего диапазона давлений. Внося незначительные изменения в нормировку данных и учитывая зависимость давления от временного шага, мы смогли построить одну нейронную сеть, описывающую

всю модель, не потеряв при этом ни скорости работы, ни качества результата.

## ЛИТЕРАТУРА

1. Peng W. Y., Pinkowski N. H. Efficient and accurate time-integration of combustion chemical kinetics using artificial neural networks. — 2017. — <https://cs229.stanford.edu/proj2017/final-reports/5241836.pdf>.
2. Sharma A. J., Johnson R. F., Kessler D. A., Moses A. Deep learning for scalable chemical kinetics // Proc. AIAA Scitech 2020 Forum, Orlando, FL, USA, 6–10 Jan., 2020. — AIAA 2020-0181. — DOI: 10.2514/6.2020-0181.
3. Ji W., Deng S. KiNet: A deep neural network representation of chemical kinetics. — 2021. — <https://arxiv.org/abs/2108.00455>.
4. Betelin V. B., Kryzhanovsky B. V., Smirnov N. N., Nikitin V. F., Karandashev I. M., Malsagov M. Yu., Mikhailchenko E. V. Neural network approach to solve gas dynamics problems with chemical

- transformations // Acta Astronaut. — 2020. — V. 180. — P. 58–65. — DOI: 10.1016/j.actaastro.2020.11.058.
5. **Nikitin V. F., Karandashev I. M., Malsagov M. Yu., Mikhalchenko E. V.** Approach to combustion calculation using neural network // Acta Astronaut. — 2022. — V. 194. — P. 376–382. — DOI: 10.1016/j.actaastro.2021.10.034.
  6. **GRI-Mech** Version 3.0 7/30/99 CHEMKINII format. — [http://www.me.berkeley.edu/gri\\_mech/](http://www.me.berkeley.edu/gri_mech/).
  7. **Новиков Е. А.** Исследование (M,2)-методов решения жестких систем // Вычислит. технол. — 2007. — Т. 12, № 5. — С. 103–115.
  8. **Ronneberger O., Fischer P., Brox T.** U-net: Convolutional networks for biomedical image segmentation // Int. Conf. on Medical Image Computing and Computer-assisted Intervention. — Springer, Cham, 2015. — P. 234–241. — DOI: 10.1007/978-3-319-24574-4\_28.

*Поступила в редакцию 25.10.2022.  
Принята к публикации 09.11.2022.*

---