

УДК 004.93'11

DOI: [10.26102/2310-6018/2025.51.4.039](https://doi.org/10.26102/2310-6018/2025.51.4.039)

Анализ методов улучшения сходимости триплетной функции потерь в многометочной классификации

Я.С. Петрова 

*Московский государственный технический университет им. Н.Э. Баумана, Москва,
Российская федерация*

Резюме. Статья посвящена улучшению результатов многометочной классификации изображений за счет улучшения сходимости триплетной функции с гибкой границей. Для этого проанализированы модификации самой функции потерь и процесса обучения, которые помогают стабилизировать градиентный спуск и достичь минимума функции потерь. Из них были выявлены перспективные гипотезы для экспериментальной проверки: заимствование способа расчета расстояния из фокальной функций потерь, замена линейного роста параметра границы между классами на логарифмический, балансировка классов в батче и корректировка его размера. Предлагаемые методы улучшения сходимости были проверены на открытом датасете CIFAR с иерархическими метками. Эффективность выбранных методов была подтверждена: каждое из изменений увеличивает точность итоговой модели на 2–4 %, при внедрении всех изменений точность классификации возросла на 10 % для классов нижнего уровня и 12 % для классов верхнего уровня. Предлагаемые методы были также проверены на устойчивость к ошибкам в датасете. Было показано, что метод улучшения сходимости за счет балансировки классов в батче чувствителен к ошибкам, поэтому не рекомендован к использованию для зашумленных датасетов.

Ключевые слова: многометочная классификация, компьютерное зрение, функция потерь, триплеты, метрическое обучение, оптимизация.

Для цитирования: Петрова Я.С. Анализ методов улучшения сходимости триплетной функции потерь в многометочной классификации. *Моделирование, оптимизация и информационные технологии*. 2025;13(4). URL: <https://moitvvt.ru/ru/journal/pdf?id=2094> DOI: 10.26102/2310-6018/2025.51.4.039

Analysis of methods for improving the convergence of triplet loss in multi-label classification

I.S. Petrova 

Bauman Moscow State Technical University, Moscow, the Russian Federation

Abstract. The paper focuses on improving the performance of multi-label image classification by enhancing the convergence of the triplet loss function with a flexible margin. To achieve this, several modifications to the loss function itself and the training process were analyzed, aiming to stabilize gradient descent and reach the loss function minimum more efficiently. Based on the analysis, several promising hypotheses were identified for experimental evaluation: adopting the distance computation approach from the focal loss, replacing the linear increase of the margin parameter with a logarithmic one, balancing classes within a batch, and adjusting batch size. The proposed convergence improvement methods were tested on the open-source CIFAR dataset with hierarchical labels. The effectiveness of the selected methods was confirmed: each modification increased the final model accuracy by 2–4%, while applying all of them together improved classification accuracy by 10% for lower-level labels and 12% for higher-level labels. The proposed methods were also evaluated for robustness to dataset noise. It was shown that the convergence improvement method based on batch class balancing is sensitive to data errors and is therefore not recommended for use with noisy datasets.

Keywords: multi-label classification, computer vision, loss function, triplets, metric learning, optimization.

For citation: Petrova I.S. Analysis of methods for improving the convergence of triplet loss in multi-label classification. *Modeling, Optimization and Information Technology*. 2025;13(4). (In Russ.). URL: <https://moitvvt.ru/ru/journal/pdf?id=2094> DOI: 10.26102/2310-6018/2025.51.4.039

Введение

Многометочная классификация – актуальная задача компьютерного зрения в связи с появлением все большего количества объемных, но неточно размеченных датасетов. Изображению может соответствовать несколько меток, при этом не все из них одинаково важны. Помимо этого, многометочная классификация позволяет привнести в пространство признаков модели «человеческое» видение: иерархии и группы классов, которые схожи по своей сути, даже если отличаются визуально. С развитием больших мультимодальных моделей сохраняется потребность в небольших моделях для классификации, так как они требуют значительно меньше вычислительных ресурсов и могут запускаться на устройстве, что критично для конфиденциальных данных и при отсутствии постоянного интернет-соединения.

Для моделирования пространства признаков изображений в задачах классификации отлично подходит метрическое обучение. В отличие от кросс-энтропии оно позволяет более гибко подходить к многометочной классификации и дает полезные эффекты в виде добавления новых классов без дообучения и возможности визуализации, кластеризации полученного пространства. В метрическом обучении для классификации себя зарекомендовала триплетная функция потерь.

В многометочной классификации есть два основных варианта: метки могут составлять иерархию (не пересекаться) или группы (возможны пересечения). Оба можно смоделировать с помощью функции потерь с гибкой границей [1], причем ее можно успешно применять не только для классификации изображений, но и текстов [2]. Однако внедрение гибкой границы негативно влияет на сходимость модели, так как градиенты функции потерь становятся менее стабильными, обучение усложняется. В [1] было показано, что настройка стратегии подбора триплетов стабилизирует процесс обучения, однако и это не гарантирует достижения оптимальных весовых коэффициентов модели.

Цель данной статьи – улучшить результаты многометочной классификации изображений за счет улучшения сходимости триплетной функции с гибкой границей. Для этого необходимо:

- 1) проанализировать модификации самой функции потерь и процесса обучения, которые могут влиять на достижение минимума функции;
- 2) выявить из них перспективные и экспериментально проверить их в реальных задачах многометочной классификации.

Материалы и методы

Способы улучшения сходимости можно разделить на три категории:

- 1) модификации функции потерь;
- 2) изменения в стратегии выбора мини-батча для обучения;
- 3) коррекция прочих гиперпараметров обучения.

Модификации функции потерь. Сама триплетная функция потерь достаточно прозрачна даже в многометочной версии с гибкой границей M , как в формуле ниже:

$$L(a, p, n, M) = \max(0, \|a - p\|^2 - \|a - n\|^2 + M). \quad (1)$$

Здесь a, p, n – векторы якорного изображения, положительного (того же класса) и отрицательного (другого класса) соответственно. Так как параметр границы не зависит от весов модели, в процессе обучения при вычислении градиента функции потерь L по матрице весов модели W он устраняется:

$$\nabla L = 2 \left[(n - p) \frac{\partial a}{\partial w} + (p - a) \frac{\partial p}{\partial w} + (a - n) \frac{\partial n}{\partial w} \right]. \quad (2)$$

Таким образом, внедрение гибкой границы влияет на сходимость только в том, что регулирует количество триплетов, при которых значение функции потерь и ее производная не равны нулю. Это, в свою очередь, влияет на значение градиента на каждом шаге, а значит может вызывать нестабильность в обучении, если эта величина слишком резко изменяется. Чем больше значение M , тем больше нерешенных [3] триплетов. Когда значение M сильно изменяется от батча к батчу или для разных триплетов внутри батча, это может вести к нестабильному значению градиента функции потерь: он то рассчитывается на почти тривиальных триплетах, то на сложных, то усредняется на большом количестве примеров, то на нескольких.

В исходной формуле триплетной функции потерь с гибкой границей параметр M выбирается в зависимости от того, сколько меток классов не совпадает у двух объектов. Это может быть или выбор из заранее заданного списка вида: 1 совпадение – m_1 , 2 совпадения – $m_2 \dots$ и так далее. Или константа m , умножаемая на количество не совпавших меток. В первом случае разработчик полностью контролирует выбор границы, во втором она растет линейно. Однако линейный рост слишком кардинален для обучения, и это усугубляется при увеличении глубины иерархии классов. Так, в пределах одного батча градиент может отличаться в 5–10 раз, что тоже негативно влияет на градиентный спуск.

Чтобы гибкая граница не дестабилизировала обучение, предлагается использовать логарифмический рост границы как при задании ее вручную, так и если она умножается на количество совпавших меток. Тогда, если n – количество не совпавших меток, а m – константное значение границы, то вместо формулы:

$$M(n) = n \cdot m, \quad (3)$$

предлагается использовать:

$$M(n) = m \cdot \log n. \quad (4)$$

В таком случае рост границы при увеличении количества не совпавших меток будет менее радикальным, а изменения градиента между батчами – более стабильным.

В формулах (3) и (4) параметр n так же можно настроить как количество совпавших меток, тогда граница будет тем выше, чем меньше различий между метками объектов. Такой подход позволяет качественнее разделить похожие объекты на нижнем уровне иерархии.

Еще один способ достигнуть оптимальных весовых коэффициентов модели – использовать не классическую триплетную функцию потерь, а ее модификацию – триплетную фокальную функцию [4]. В ней способ подсчета расстояний корректируется, приводя формулу классической триплетной функции к виду:

$$L(a, p, n) = \max(0, e^{\frac{\|a-p\|^2}{\sigma}} - e^{\frac{\|a-n\|^2}{\sigma}} + \alpha). \quad (5)$$

Соответственно для формулы с гибкой границей:

$$L(a, p, n, M) = \max(0, e^{\frac{\|a-p\|^2}{\sigma}} - e^{\frac{\|a-n\|^2}{\sigma}} + M). \quad (6)$$

Основная мотивация внедрения экспоненты в расчет расстояний – усилить влияние сложных триплетов, в которых расстояние между якорным изображением и положительным велико, а между якорным и отрицательным – мало. Введение $\sigma \in (0; 1)$ гиперболизирует большие расстояния и незначительно влияет на малые. Соответственно вклад сложных триплетов в итоговую функцию будет больше, рассчитанные градиенты – более полезны. С точки зрения стабильности важно выбирать σ не слишком малым, к примеру, $\sigma \in [0,5; 1)$, так как выбор меньших значений приводит к более сильному эффекту фокусировки на сложных триплетах, а значит и более резким изменениям в значениях градиентов между батчами.

В результате для повышения эффективности обучения в многометочных задачах предлагается использовать триплетную фокальную функцию с гибкой логарифмической границей. Границу следует выбирать пропорциональной количеству совпавших меток.

Изменения в стратегии выбора мини-батча для обучения. Помимо изменений в функции потерь, повлиять на эффективность процесса обучения можно и за счет подбора примеров в батче. Есть популярные стратегии для триплетных функций: на каждом шаге использовать самые сложные или полусложные триплеты. В исследовании [1] было показано, что использование смеси этих стратегий помогает стабилизировать обучение и избежать тривиального решения, когда модель проецирует векторы всех изображений в одну точку, тем самым достигая локального минимума равного значению параметра границы. Также существует стратегия выбора простых положительных пар [5], в которой предлагается выбирать сложные отрицательные пары, но простейшие положительные, ведь векторы одного класса не обязательно должны стягиваться в одну точку пространства, а значит нет необходимости сближать изначально далеко расположенные экземпляры одного класса. Так как доминирующей стратегией при обучении является выбор полусложных триплетов, в которых и так не выбираются сложные положительные пары, этот нюанс уже учтен в стандартном процессе обучения.

Однако сила влияния стратегии выбора зависит от размера мини-батча, особенно при выборе самых сложных триплетов. Если батч слишком большой, то в нем выше вероятность найти аномально сложные триплеты. Тогда веса модели будут изменяться так, чтобы правильно расположить эти редкие выбивающиеся примеры в векторном пространстве, даже если при этом нарушается общая структура. В случае выбора слишком маленького батча в нем может не оказаться достаточного количества сложных триплетов, и тогда градиенты будут рассчитаны на одном-двух примерах, что так же не ведет к увеличению эффективности обучения.

Поэтому предлагается выбирать средний размер батча (128–512) примеров, в зависимости от количества классов. Чем больше возможных меток, тем больше батч понадобится для подбора оптимальных триплетов. При этом для всех классов должно присутствовать хотя бы по 2 экземпляра, чтобы объект каждого класса гарантированно хотя бы раз был выбран в качестве якорного в триплете. Такое семплирование также решает проблему дисбаланса классов в датасете, типичную для многометочных задач [6].

Существует несколько стратегий подбора батча для повышения эффективности обучения в многометочных задачах. Большинство из них предназначены для не иерархической классификации, где метки разрежены и их много [7]. Поэтому многие подходы построены на разных способах взвешивания пар, чтобы отобрать максимально информативные триплеты, например, выбор максимально отличающихся якорных элементов [8], учет степени сходства меток [9, 10] или степени важности конкретной метки [11]. Однако в данном исследовании степень влияния сходства и важности меток уже учтена в гибкой границе функции потерь, поэтому реализация данных подходов избыточна. Они могут быть рассмотрены в дальнейших исследованиях в рамках

улучшения сходимости обучения с разреженными метками на неточно размеченных датасетах.

Также существует популярный подход [12, 13], где для многометочного обучения смещается количество положительных пар в триплетах. При обычной многоклассовой классификации положительной парой считаются два изображения с полностью совпадающими классами. Однако при многометочном обучении с таким подходом положительных пар может оказаться слишком мало или не найтись вообще, так как даже одна различающаяся метка делает пару отрицательной. Авторы предлагают считать положительными пары объектов, у которых совпадает хотя бы одна метка. К сожалению, такой подход не универсален для любых многометочных задач: при иерархической классификации функция потерь утратит способность разграничивать объекты на нижнем уровне иерархии, ведь у них всегда будут одинаковые метки на верхних уровнях.

Еще один возможный метод подбора примеров в батч – обучение «от общего к частному». На первых эпохах батч составляется из примеров, отличающихся на верхнем уровне иерархии классов, чтобы примерно организовать векторное пространство. Дальше подбираются примеры, отличающиеся на нижних уровнях иерархии, чтобы окончательно сформировать векторы. У триплетной функции потерь этот эффект присутствует и без явного моделирования батча за счет стратегий выбора самых сложных и полусложных примеров.

В результате анализа имеющихся методов улучшения сходимости применительно к триплетной функции, предлагается выбирать размер батча сравнительно небольшим (128–512 экземпляров) и составлять его из примеров каждого класса.

Коррекция прочих гиперпараметров обучения. Помимо состава мини-батчей и функции потерь на эффективность обучения значительно влияет скорость обучения. Существуют две популярные стратегии изменения скорости обучения: разогрев (англ. warm-up) и угасание (англ. decay). Разогрев – это постепенное наращивание скорости от 0 до целевой в начале обучения. В начале веса модели очень далеки от оптимальных, поэтому если сразу использовать слишком большую скорость, направление градиентов будет сложно стабилизировать. Угасание – это постепенное уменьшение скорости с течением обучения, так как чем ближе веса к оптимальным, тем более точная коррекция им требуется. Однако при детальном исследовании разогрева [14] выяснилось, что он оказывает положительное влияние только на некоторые функции потерь. Так же его положительное влияние сводится к тому, что чем дольше длится период разогрева, тем допустимее становятся большие значения скорости обучения. Поэтому разогрев может влиять на скорость обучения, но не на успешность оптимизации.

Если рассмотреть обе стратегии применительно к триплетной функции, то их внедрение, напротив, может навредить. В начале обучения триплетная функция быстро организует пространство, используя большие значения градиентов. Если внедрить в этот этап разогрев, это снизит эффективность формирования векторного пространства. Ближе к концу обучения нерешенных триплетов становится мало и градиенты естественным образом уменьшаются, поэтому применение угасания также избыточно.

Результаты

Для экспериментальной проверки эффективности предложенных изменений в функцию потерь и способа формирования батча был использован датасет CIFAR. Это бенчмарк, содержащий 60000 изображений, разделенных на 100 классов и 20 категорий. Соответственно каждому изображению соответствует две метки – класс и категория, поэтому датасет подходит для проверки гипотез в многометочной классификации. В качестве модели использовалась FaceNet-подобная сверточная сеть для классификации

изображений 32×32 пикселя. Она состояла из 4 сверточных блоков, в каждом: сверточный слой (3×3), слой пакетной нормализации, слой понижения размерности (2×2) со взятием максимума. Количество фильтров свертки с каждым блоком возрастало в два раза: от 64 до 512. В завершении 2 полносвязных слоя: 256 и 128 нейронов, к последнему применялась L2-нормализация для ограничения длины векторных представлений.

Было проведено несколько экспериментов:

1. Обучение с триплетной функцией потерь и триплетной с гибкой границей – параметр границы задается пропорционально количеству не совпавших меток.

2. Обучение с теми же функциями, но расстояния в них рассчитываются по правилам фокальной функции.

3. Обучение с триплетной функцией с гибкой границей (оба варианта), фокальными расстояниями и логарифмическим ростом (или снижением) границы.

4. Обучение, как в предыдущем пункте, но еще и со сбалансированными батчами среднего размера (300).

Во всех экспериментах использовалась стратегия выбора полусложных триплетов. Валидационная функция ошибки – триплетная со стратегией выбора самых сложных триплетов. Результаты экспериментов приведены в Таблицах 1 и 2.

Таблица 1 – Значение функции ошибки на валидационном датасете CIFAR после обучения

Table 1 – The value of the loss function on the CIFAR validation dataset after training

Функция потерь	Стандартный процесс обучения	С фокальными расстояниями	С фокальными расстояниями и логарифмическим изменением границы	С фокальными расстояниями и логарифмическим изменением границы и сбалансированным батчем
Триплетная функция	1,58	1,46	–	–
Триплетная функция с гибкой границей	1,52	1,43	1,41	1,38

Таблица 2 – Точность распознавания классов и категорий CIFAR после обучения

Table 2 – The classification accuracy of CIFAR classes and categories after training

Функция потерь	Стандартный процесс обучения, %	С фокальными расстояниями, %	С фокальными расстояниями и логарифмическим изменением границы, %	С фокальными расстояниями и логарифмическим изменением границы и сбалансированным батчем, %
Триплетная функция	42 (57)	46 (61)	–	–
Триплетная функция с гибкой границей	46 (60)	48 (64)	50 (64)	52 (69)

Точность обученных моделей была рассчитана по формуле макро F1. В таблице она указана в формате: класс (категория).

Также в датасет был добавлен 1 % ошибок, чтобы проверить устойчивость модели к шуму. Оценка точности моделей, обученных на датасете, с ошибками приведена в Таблице 3.

Таблица 3 – Точность распознавания классов и категорий CIFAR после обучения на датасете с ошибками

Table 3 – The classification accuracy of CIFAR classes and categories after training on the noisy dataset

Функция потерь	Стандартный процесс обучения, %	С фокальными расстояниями, %	С фокальными расстояниями и логарифмическим изменением границы, %	С фокальными расстояниями и логарифмическим изменением границы и сбалансированным батчем, %
Триплетная функция	41 (57)	43 (58)	–	–
Триплетная функция с гибкой границей	45 (60)	46 (61)	46 (62)	46 (62)

Обсуждение

Экспериментальная проверка показала, что добавление фокальных расстояний, логарифмического роста/убывания параметра границы и корректировка батча помогают достичь меньшего значения функции потерь на валидационном датасете. При использовании всех модификаций, значение функции потерь уменьшилось на 13 %. Это привело к повышению точности распознавания итоговой модели. Каждая модификация добавляет 2–4 % к точности, суммарно точность увеличилась на 10 % для классов и на 12 % для категорий.

Триплетная функция с гибкой границей проявляет такую же устойчивость к ошибкам, как и оригинальная. То же можно сказать и о фокальных расстояниях. При внедрении логарифма в расчет границы устойчивость к ошибкам снижается. Особенно заметно снижение устойчивости к шуму при применении сбалансированного батча. Это объясняется тем, что некоторые малочисленные классы теперь содержат ошибки и, так как в батче классы сбалансированы, оказывают более существенное влияние на результат. Поэтому балансировку батча следует применять с осторожностью на зашумленных датасетах.

Заключение

В результате исследования методов улучшения сходимости были предложены следующие изменения в процессе обучения:

1. Логарифмический рост гибкой границы вместо изначального линейного.
2. Расчет расстояний внутри триплета по правилам фокальных функций.
3. Балансировка классов в батче, выбор среднего (128–512) размера батча.

Экспериментальная проверка предложенных изменений на датасете CIFAR показала, что с их использованием точность итоговой модели может быть улучшена на 10 % для классов и 12 % для категорий. Проверка устойчивости к ошибкам показала, что

балансировку классов нежелательно использовать на зашумленных датасетах. Остальные изменения не повлияли на устойчивость обучения к шуму.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Петрова Я.С. Методика обучения классификаторов изображений с использованием дополнительных меток. *Моделирование, оптимизация и информационные технологии*. 2025;13(2). <https://doi.org/10.26102/2310-6018/2025.49.2.041>
Petrova I.S. Method of Training Image Classifiers Using Additional Labels. *Modeling, Optimization and Information Technology*. 2025;13(2). (In Russ.). <https://doi.org/10.26102/2310-6018/2025.49.2.041>
2. Deng W., Zhang J., Zhang P., Yao Y., Gao H., Zhang Y. Hyper-Label-Graph: Modeling Branch-Level Dependencies of Labels for Hierarchical Multi-Label Text Classification. In: *ACML 2023: Asian Conference on Machine Learning, 11–14 November 2023, Istanbul, Turkey*. PMLR; 2023. P. 279–294.
3. Сулоев К.К., Шешкус А.В., Арлазаров В.Л. Сферические ограничения в триплетной функции потерь. *Труды Института системного анализа Российской академии наук*. 2023;73(2):50–58. <https://doi.org/10.14357/20790279230205>
Suloev K.K., Sheshkus A.V., Arlazarov V.L. Spherical Constraints in the Triplet Loss Function. *Proceedings of the Institute for Systems Analysis Russian Academy of Sciences*. 2023;73(2):50–58. (In Russ.). <https://doi.org/10.14357/20790279230205>
4. Zhang Sh., Zhang Q., Wei X., Zhang Y., Xia Y. Person Re-Identification with Triplet Focal Loss. *IEEE Access*. 2018;6:78092–78099. <https://doi.org/10.1109/ACCESS.2018.2884743>
5. Xuan H., Stylianou A., Pless R. Improved Embeddings with Easy Positive Triplet Mining. In: *2020 IEEE Winter Conference on Applications of Computer Vision (WACV), 01–05 March 2020, Snowmass, CO, USA*. IEEE; 2020. P. 2463–2471. <https://doi.org/10.1109/WACV45572.2020.9093432>
6. Suyal H., Shivhare Sh.N., Shrivastava G., Singh R., Singhal A. IA-KNNR: A Novel Imbalance-Aware Approach for Handling Multi-Label Class Imbalance Problem. *IEEE Access*. 2025;13:119999–120017. <https://doi.org/10.1109/ACCESS.2025.3586146>
7. Liu W., Wang H., Shen X., Tsang I.W. The Emerging Trends of Multi-Label Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2021;44(11):7955–7974. <https://doi.org/10.1109/TPAMI.2021.3119334>
8. Sumbul G., Ravanbakhsh M., Demir B. Informative and Representative Triplet Selection for Multilabel Remote Sensing Image Retrieval. *IEEE Transactions on Geoscience and Remote Sensing*. 2021;60. <https://doi.org/10.1109/TGRS.2021.3124326>
9. Wang Ch., Liu F., Chen Y., Frazer H., Carneiro G. Cross- and Intra-Image Prototypical Learning for Multi-Label Disease Diagnosis and Interpretation. *IEEE Transactions on Medical Imaging*. 2025;44(6):2568–2580. <https://doi.org/10.1109/TMI.2025.3541830>
10. Wang X., Han X., Huang W., Dong D., Scott M.R. Multi-Similarity Loss with General Pair Weighting for Deep Metric Learning. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 15–20 June 2019, Long Beach, CA, USA*. IEEE; 2019. P. 5017–5025. <https://doi.org/10.1109/CVPR.2019.00516>
11. Shao R., Xu N., Geng X. Multi-Label Learning with Label Enhancement. In: *2018 IEEE International Conference on Data Mining (ICDM), 17–20 November 2018, Singapore*. IEEE; 2018. P. 437–446. <https://doi.org/10.1109/ICDM.2018.00059>
12. Kharbanda S., Gupta D., K G., et al. UniDEC: Unified Dual Encoder and Classifier Training for Extreme Multi-Label Classification. In: *WWW '25: Proceedings of the ACM on Web Conference 2025, 28 April – 02 May 2025, Sydney, Australia*. New York:

- Association for Computing Machinery; 2025. P. 4124–4133. <https://doi.org/10.1145/3696410.37147>
13. Csányi G.M., Vági R., Megyeri A., et al. Can Triplet Loss Be Used for Multi-Label Few-Shot Classification? A Case Study. *Information*. 2023;14(10). <https://doi.org/10.3390/info14100520>
 14. Kalra D.S., Barkeshli M. Why Warmup the Learning Rate? Underlying Mechanisms and Improvements. In: *NeurIPS 2024: Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, 10–15 December 2024, Vancouver, BC, Canada*. 2024. P. 111760–111801. <https://doi.org/10.5202/079017-3549>

ИНФОРМАЦИЯ ОБ АВТОРЕ / INFORMATION ABOUT THE AUTHOR

Петрова Яна Сергеевна, ассистент, **Iana S. Petrova**, Assistant, Bauman Moscow Московский государственный технический State Technical University, Moscow, the Russian университет им. Н.Э. Баумана, Москва, Federation.

Российская Федерация.

e-mail: ypetrova@bmstu.ru

ORCID: [0009-0005-4377-7981](https://orcid.org/0009-0005-4377-7981)

Статья поступила в редакцию 06.10.2025; одобрена после рецензирования 07.11.2025; принята к публикации 14.11.2025.

The article was submitted 06.10.2025; approved after reviewing 07.11.2025; accepted for publication 14.11.2025.