

М. М. Липкович, канд. физ.-мат. наук, ст. науч. сотр., lipkovich.mikhail@gmail.com,
Институт проблем машиноведения РАН, г. Санкт-Петербург,
А. Р. Сагатдинов, студент, amazar226@gmail.com,
Санкт-Петербургский государственный университет

Применение алгоритма "Полоска" для онлайн-декодирования ЭЭГ-паттернов*

Рассматривается задача определения по сигналам электроэнцефалограммы того, какой рукой — правой или левой — испытуемый намеревается совершить движение. Актуальность задачи обусловлена широким распространением интерфейсов мозг—компьютер, где электроэнцефалография является одним из основных неинвазивных методов снятия сигналов с головного мозга. Определив руку, которой испытуемый намеревается совершить движение, можно подавать соответствующие команды в компьютер.

Для решения поставленной задачи из отрезков сигналов, предшествующих движению, выделяются временные и частотные признаки, которые подаются на вход модели машинного обучения. Задача формулируется как задача бинарной классификации. Модель должна предсказать, будет ли движение совершено правой рукой или левой.

В отличие от стандартной постановки задачи обучения с учителем предполагается, что предзаданный набор данных для обучения отсутствует, и семплы для обучения модели поступают один за другим. Таким образом имитируется ситуация, при которой модель должна работать с новым испытуемым и подстроиться под него в реальном времени. Традиционным методом обучения линейных моделей в такой парадигме является стохастический градиентный спуск. Ранее было показано, что алгоритм "Полоска", разработанный В. А. Якубовичем для ряда задач, обладает более высокой скоростью сходимости, чем стохастический градиентный спуск. Однако это достигается за счет того, что шаг алгоритма совершается на каждый признак семпла. Это делает рассматриваемую ранее версию "Полоски" не пригодной для работы с данными высокой размерности. В статье предлагается использование другой версии "Полоски", не обладающей указанным недостатком.

Предложенный алгоритм апробирован на открытом наборе данных с соревнований "BCI competition II". Показано, что алгоритм обладает более высокой скоростью совершения одного шага обучения по сравнению с традиционными линейными моделями на основе стохастического градиентного спуска на рассматриваемом наборе данных, что является преимуществом при использовании в реальном времени.

Ключевые слова: рекуррентные целевые неравенства, интерфейсы мозг—компьютер, классификация ЭЭГ, машинное обучение

Введение

В настоящее время широкое распространение получили интерфейсы мозг—компьютер [1], применяемые в протезировании [2], реабилитации [3], лечении неврологических заболеваний [4, 5] и других областях. Для обеспечения взаимодействия между мозгом и компьютером необходимо снимать сигналы с мозга. Одним из наиболее популярных неинвазивных методов снятия сигналов с мозга является электроэнцефалография (ЭЭГ) [6]. В случае использования ЭЭГ возникает задача декодирования паттернов в целях выделения из них полезных команд. Например, определив по сигналам ЭЭГ движения правой и левой рукой, можно сопоставить им разные управляющие воздействия на мобильного робота [7]. Однако большой интерес для интерфейсов мозг—компьютер представляет задача выявления паттернов из сигналов до осуществления испы-

туемого движения [8, 9]. Таким образом, требуется определить намерение совершить движение. Данную задачу можно сформулировать как классификационную задачу машинного обучения, где требуется обучить модель, которая по признакам, выделенным из отрезков сигналов ЭЭГ, предшествующих движению, должна определять, будет ли совершено движение или какое именно движение будет совершено.

Целью данной работы является разработка алгоритма для решения задачи определения руки, которой испытуемый намеревается совершить движение. В качестве источника данных для апробации подхода используется открытый датасет "IV" с соревнований "BCI competition II"¹, в котором представлены сигналы ЭЭГ испытуемого, совершающего нажатия на клавиши пальцами левой и правой руки. Результаты классификационных моделей на основе этого набора данных представлены в различных

*Исследование выполнено за счет гранта Российского научного фонда (проект № 21-71-00144).

¹https://www.bbci.de/competition/ii/berlin_desc.html

статьях [10, 11]. Особенностью данной работы является рассмотрение поставленной задачи с позиции онлайн-машинного обучения: будем предполагать, что предзаданный тренировочный набор для обучения отсутствует. Будем считать, что семплы поступают в модель по одному, и требуется в реальном времени как можно быстрее подстроить модель под поступающие семплы. Таким образом, имитируется ситуация, при которой интерфейсом будет пользоваться новый испытуемый, по которому предварительно собранные данные отсутствуют. Испытуемый будет совершать некоторое число движений согласно инструкциям от программы, вследствие чего у модели будут появляться семплы для обучения. Требуется настроить параметры модели, используя минимальное число итераций обучения. Вопросы применения интерфейсов мозг—компьютер в реальном времени были рассмотрены в работах [12, 13], однако в них не предполагалось обучения моделей в процессе использования интерфейса.

Предложенный алгоритм состоит из трех шагов: предобработка данных, выделение признаков и обучение классификационной модели. Будут выделяться две категории признаков: временные и частотные. Для каждой категории признаков будет применяться своя предобработка, состоящая из центрирования сигналов и фильтров. В качестве классификационной модели будет рассматриваться алгоритм "Полоска", разработанный В. А. Якубовичем [14]. Ранее было показано, что алгоритм "Полоска" для ряда задач обладает более быстрой сходимостью по сравнению со стохастическим градиентным спуском [15, 16], являющимся стандартным методом для обучения линейных моделей в парадигме он-

лайн-машинного обучения. Однако модель из работы [16] обладает недостатком для данных высокой размерности: на каждый семпл данных требуется осуществить число итераций, пропорциональное размерности признаков. Для преодоления этого ограничения в данной работе рассматривается вариация алгоритма "Полоска", названная В. А. Якубовичем "С-системой" [17]. Эта версия алгоритма совершает один шаг настройки параметров на один семпл вне зависимости от числа признаков. В работе проведен сравнительный анализ алгоритмов "С-система" и "Полоска", а также алгоритма стохастического градиентного спуска в приложении к задаче определения того, какой рукой испытуемый намеревается совершить движение.

Статья организована следующим образом: в разделе "Набор данных" описаны данные, используемые для апробации предложенного алгоритма. В разделе "Выделение признаков" описаны методы, используемые для предобработки сигналов и выделения признаков из рассматриваемого набора данных. Раздел "Алгоритм "Полоска" посвящен описанию модели, используемой для классификации сигналов и, наконец, в разделе "Результаты" описаны результаты численного сравнения алгоритма "Полоска" со стохастическим градиентным спуском.

Набор данных

В качестве набора данных используется датасет "IV" из "BCI competition II" [18]. Датасет был собран при исследовании здорового испытуемого с использованием 28 Ag/AgCl-электродов с частотой дискретизации 1000 Гц

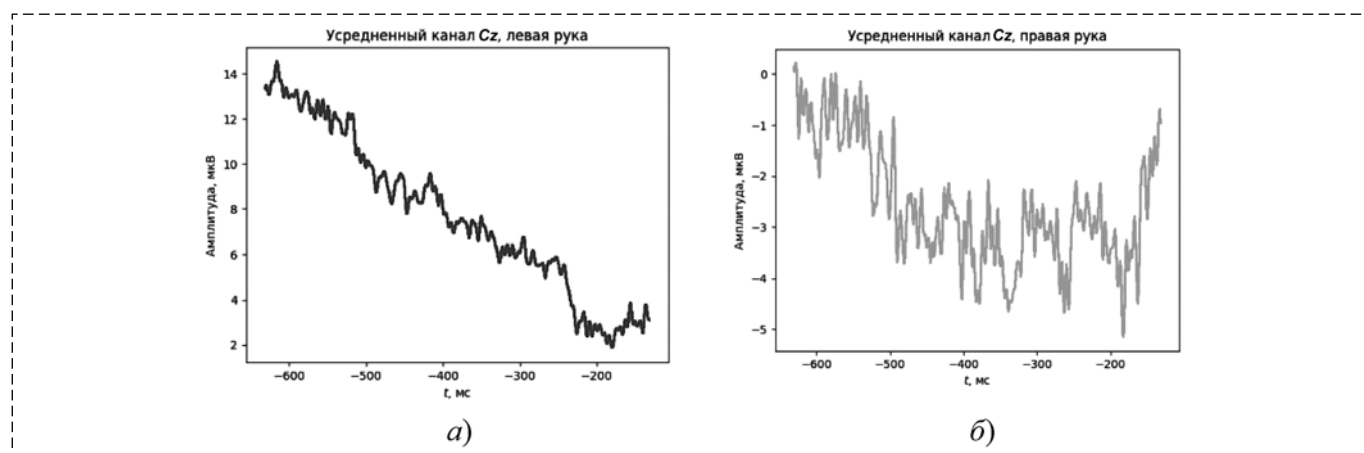


Рис. 1. Усредненные амплитуды по каждому из классов канала Cz

Fig. 1. Averaged amplitudes of each class for channel Cz

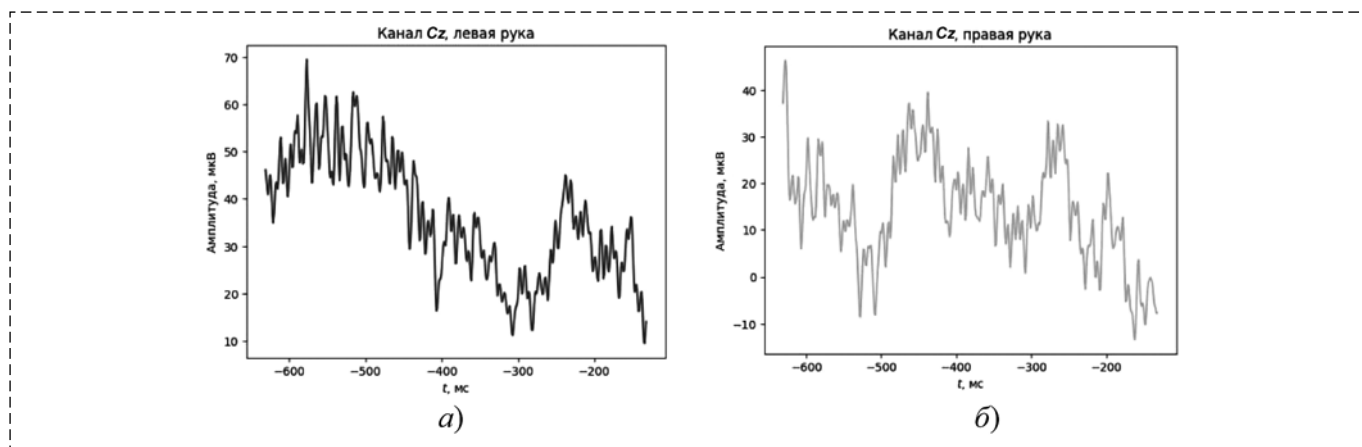


Рис. 2. Примеры индивидуальных положительных (а) и отрицательных (б) эпох канала Cz
 Fig. 2. Examples of individual positive (a) and negative (b) epochs for channel Cz

и полосным фильтром 0,05...200 Гц. Датасет включает в себя 416 эпизодов движений (эпох). Каждая эпоха имеет длину 500 мс и заканчивается за 130 мс до совершения движения. Движения представляют собой нажатия на клавишу. В каждой эпохе используются относительные значения времени, момент нажатия считается нулевым моментом времени. Таким образом, каждая эпоха начинается в моменте времени -630 мс и заканчивается на -130 мс.

Рассматриваем два класса эпох — *положительные*, где движение было выполнено левой рукой, и *отрицательные*, где движение было выполнено правой рукой. Число положительных и отрицательных эпох совпадает. На рис. 1 приведены усредненные амплитуды сигнала для положительных (рис. 1, а) и отрицательных (рис. 1, б) эпох в канале Cz, соответствующему центральному вертексному электроду.

Как видно из графиков, при положительных эпохах наблюдается более ярко выраженное снижение амплитуды. Однако распознавание должно осуществляться не по усредненным по эпохам данным, а по индивидуальным, где в силу низкого соотношения сигнал—шум различия не столь ярко выражены. Для примера на рис. 2 приведены амплитуды сигнала для одной эпохи каждого из классов того же канала.

Выделение признаков

Как видно из рис. 2, различия в индивидуальных эпохах могут быть не столь заметны. Необходимо выделять признаки, которые будут подаваться на вход бинарной модели классифи-

кации в целях определения того, какой рукой испытуемый намеревается совершить движение.

Признаки выделялись как во временной, так и в частотной областях. Сигналы подвергались при этом разным вариантам предобработки.

Предобработка для временных признаков

Каждый канал ЭЭГ был подвергнут следующей процедуре:

1. Удаление глобального среднего для удаления смещения изолинии от нуля.
2. Фильтрация полосовым фильтром с частотами среза 0,3 Гц и 3 Гц. Для этого использовался КИХ-фильтр, синтезированный методом окон с окном Хемминга. Порядок фильтра был равен 101.
3. Удаление участков сигнала с амплитудами выше 500 мкВ. При этом вокруг каждого семпла, превышающего эту величину, дополнительно удалялся участок в 200 мс вокруг.
4. К каналам C1, C2, C3, C4 был дополнительно применен большой лапласиановский фильтр. К каналам Cz, FC3, FC4 применялся малый лапласиановский фильтр.

Предобработка для частотных признаков

В целях улучшения качества распознавания сигналов во временной области, где не удается выделить признаки, позволяющие отделить положительные эпохи от отрицательных, дополнительно были выделены признаки в частотной области. Предобработка сигналов для этих признаков была следующей:

1. Удаление глобального среднего для удаления смещения изолинии от нуля.

2. Применение ФНЧ с частотой среза 50 Гц.
3. Удаление участков сигнала с амплитудами выше 500 мкВ. При этом вокруг каждого семпла, превышающего это значение, дополнительно удалялся участок в 200 мс вокруг.

4. Построение лапласиановских версий сигналов аналогично предобработке во временной области.

5. Вычисление вызванной синхронизации/десинхронизации:

1) к сигналу применяется оконное преобразование Фурье с окном Хеннинга длины 175 мс и сдвигом между окнами в 5 мс;

2) для каждого окна вычисляется мощность сигнала на наборе частотных полос:

- α_1 (8...9 Гц),
- α_2 (10...12 Гц),
- β_1 (13...17 Гц),
- β_2 (18...30 Гц);

3) сигнал для каждой полосы переводится во временную область, где каждому отрезку времени соответствуют мощности этой полосы в данный момент времени.

По каждой эпохе первые 175 мс используются для коррекции: по этому участку на каждом канале вычисляется среднее, и это среднее вычитается из каждого семпла канала и на него же нормируется.

Одни и те же типы признаков извлекались по отдельности из сигналов с временной предобработкой и частотной. Были использованы четыре категории признаков:

1. Признаки на основе линейной регрессии. Эти признаки строились по каналам F_z , лапласиановской версии канала C_z , на разнице лапласиановских версий каналов C_3 и C_4 , а также по аналогичным разностям пар C_1 , C_2 и FC_3 , FC_4 . По каждому каналу строились две линейные регрессии с двумя коэффициентами. Одна линейная регрессия строилась с начала эпохи до точки перегиба, а вторая — с точки перегиба до конца эпохи. Точка перегиба была подобрана автоматически исходя из условия минимизации среднеквадратичного отклонения полученных двух регрессионных прямых от исходного сигнала. Также в качестве дополнительного признака использовалось значение отступа от начала эпохи самой точки перегиба.

2. Признаки на основе амплитуд. Вычислялись максимальное и минимальное по модулю значения для лапласиановской версии канала C_z , а также для разниц лапласиановских версий каналов C_3 , C_4 , каналов C_1 , C_2 и FC_3 , FC_4 .

3. Признаки на основе площадей. По тем же каналам, что и в предыдущем пункте, вычислялись площади под кривой для всей эпохи.

4. Признаки на основе общего пространственного фильтра (*common spatial pattern, CSP*) [13]. Фильтр применялся к каналам F_1 — F_4 , F_z , FC_1 , FC_3 , FC_5 , C_1 — C_4 , CP_2 — CP_6 из полосы 18...30 Гц. Число компонент подбиралось на основе кросс-валидации и было в итоге равно трем.

Таким образом, на каждую эпоху были выделены 611 числовых признака. Признаки были стандартизированы путем вычитания среднего и деления на стандартное отклонение. Извлеченные признаки подаются на вход классификационной модели для обучения. В качестве такой модели используется алгоритм "Полоска", рассматриваемый в следующем разделе.

Алгоритм "Полоска"

В данном разделе будет сформулирована версия алгоритма "Полоска", названная в работе [17] "С-системой", а также, аналогично работе [16], будут рассмотрены практические соображения по его реализации, которые положительно сказываются на скорости обучения.

Пусть дан набор данных $\{x_i, y_i = s(x_i)\}$, $i = 1, \dots, m$, где $x_i \in X \subset R^n$ — векторы признаков, $s(x)$ — неизвестная функция из X в $\{1, -1\}$. Задана задача бинарной классификации, т. е. требуется восстановить функцию s по имеющемуся набору данных.

Предположим, что $\{a_j(x)\}$, $j = 1, \dots, N$, — полный в $L^2(X)$ набор функций, при этом верно, что $|a_j(x)| \leq \text{const}$.

В алгоритме "Полоска" строится классификатор вида

$$\sigma(x) = \text{sign} \left(\sum_{j=1}^N \theta_j a_j(x) \right) = \text{sign}(s_N(x, u)),$$

где u — вектор обучаемых параметров с элементами $\theta_1, \dots, \theta_N$.

Согласно работе [17] будем говорить о том, что опознающая система имеет форму "С-системы", если параметры u выбираются исходя из соотношений

$$|s(x_j) - s_N(x_j)| \leq \varepsilon$$

для некоторого достаточно маленького ε .

Пусть $0 < \delta \leq \varepsilon$, $0 \leq \beta \leq 2$, $\rho_j \rightarrow 0$. Алгоритм настройки весов примет следующий вид:

$$\begin{aligned} \eta_k &= s(x_k) - \sum_{j=1}^N \theta_j a_j(x_k); \\ \zeta_k &= -N^{-1} \eta_k, \text{ если } |\eta_k| \geq 2\varepsilon; \\ \zeta_k &= 2N^{-1}(\varepsilon \text{sign}(\eta_k) - \eta_k), \\ &\text{если } \varepsilon + \delta \leq |\eta_k| < 2\varepsilon; \\ \zeta_k &= -\rho_k + N^{-1}\beta(\varepsilon \text{sign}(\eta_k) - \eta_k), \\ &\text{если } \varepsilon \leq |\eta_k| < 2\varepsilon; \\ \zeta_k &= 0, \text{ если } \varepsilon < |\eta_k|; \\ \theta_j &= \theta_j + \zeta_k a_j(x_k). \end{aligned} \quad (1)$$

Согласно работе [17] алгоритм (1) является конечно-сходящимся.

Мы будем рассматривать алгоритм (1) в парадигме онлайн-ового машинного обучения, где семплы попадают для обучения по одному. В этом случае на каждый новый семпл будем осуществлять один шаг обучения. Когда накопится достаточное число семплов, будем осуществлять повторное обучение на всех накопленных семплах, пока качество не станет приемлемым.

Результаты

Проведем сравнение трех моделей: "Полоска", "С-система Полоска" и стохастический градиентный спуск. Все модели были обучены на признаках, описанных в разделе "Выделение признаков". Алгоритм сравнения следующий: всем трем моделям последовательно, по одному, предъявляются семплы из датасета. На каждом семпле определяются ошибки каждой из моделей и запоминается, сколько ошибок было сделано к данному моменту. Далее этот семпл предъявляется для шага обучения. В итоге обучение проводится на всем датасете, и определяется накопительная ошибка. После этого датасет подается снова по кругу в том же режиме. Это заикливание датасета осуществляется ввиду небольшого числа тренировочных семплов. На практике вместо этого можно подавать новые эпохи, либо, в случае ограниченности набора данных, также можно подавать заикленную тренировочную последовательность. Такой подход сравнения онлайн-овых моделей был предложен в работе [19].

В табл. 1 представлены метрики итоговых процентов ошибок после 100 прогона 1, 10 и 100 эпох.

Как видно из табл. 1, алгоритм "Полоска" сразу начинает опережать остальные методы, и по мере добавления циклов он опережает их сильнее. Это согласуется с предыдущими наблюдениями касательно более быстрой сходимости этого алгоритма [15, 16]. "Полоска С-система" начинает работать хуже остальных методов, однако после достаточно большого числа циклов он полностью выравнивается со стохастическим градиентным спуском. Преимущество "С-системы" при этом состоит в его скорости на один цикл обучения.

Время, затрачиваемое на один цикл обучения, представлено в табл. 2. Время измерялось на компьютере с процессором *AMD Ryzen 5 4500U* и 16 Гбайт оперативной памяти.

Таким образом, "Полоска С-система" показывает преимущество во времени обработки одного цикла обучения, тогда как "Полоска" сходится быстрее с точки зрения числа циклов, но затрачивает существенно больше времени на каждый цикл итераций из-за того, что совершает шаг обучения на каждый признак, и поэтому плохо подходит для задач высокой размерности [16].

Таблица 1
Table 1

Проценты накопительных ошибок
после заданного числа циклов
*Percent of accumulated errors
after a given number of training loops*

Метод обучения	Число циклов		
	Один цикл	10 циклов	100 циклов
Стохастический градиентный спуск	0,44	0,37	0,23
Полоска	0,43	0,33	0,19
Полоска С-система	0,49	0,43	0,23

Таблица 2
Table 2

Время, затрачиваемое на один цикл обучения
Time required for one training loop

Метод обучения	Время, мс
Стохастический градиентный спуск	141 ± 30
Полоска	3005 ± 173
Полоска С-система	5,29 ± 0,26

Заклучение

В данной работе были применены две версии алгоритма "Полоска" к детектированию намерения совершить движение на индивидуальных эпохах. Показано, что алгоритм "Полоска" в зависимости от реализации может сходиться быстрее, чем стохастический градиентный спуск, однако может требовать много времени на совершение каждой итерации. Версия "С-система" алгоритма "Полоска" дает такое же качество, как у стохастического градиентного спуска, но затрачивает меньше времени на каждый шаг. Это преимущество алгоритма "С-система" делает ее подходящей моделью для использования в парадигме онлайн-машинного обучения при высокой размерности признакового пространства.

Список литературы

1. Kawala-Sterniuk A., Browarska N., Al-Bakri A., Pelc M., Zygarlicki J., Sidikova M., Martinek, R., Gorzelanczyk E. J. Summary of over Fifty Years with Brain-Computer Interfaces—A Review // *Brain Sci.* 2021. Vol. 11, N. 43.
2. Vilela M., Hochberg L. Applications of brain-computer interfaces to the control of robotic and prosthetic arms // *Handbook of Clinical Neurology*, Elsevier. 2020. Vol. 168. P. 87–99.
3. Bockbrader M., Francisco G., Lee R., Olson J., Solinsky R., Boninger M. Brain Computer Interfaces in Rehabilitation Medicine // *The journal of injury, function, and rehabilitation.* 2018. Vol. 10.
4. Ruiz S., Birbaumer N., Sitaram R. Abnormal Neural Connectivity in Schizophrenia and fMRI-Brain-Computer Interface as a Potential Therapeutic Approach // *Frontiers in Psychiatry.* 2013. Vol. 4.
5. Sebastián-Romagosa M., Cho W., Ortner R., Murovec N., Von Oertzen T., Kamada K., Allison B. Z., Guger C. Brain Computer Interface Treatment for Motor Rehabilitation of Upper Extremity of Stroke Patients-A Feasibility Study // *Front Neurosci.* 2020. Vol. 14.
6. McFarland D. J., Wolpaw J. R. EEG-Based Brain-Computer Interfaces // *Current opinion in biomedical engineering.* 2017. Vol. 4. P. 194200.

7. Plotnikov S. A., Lipkovich M., Semenov D. M., Fradkov A. L. Artificial intelligence based neurofeedback // *Cybernetics and Physics.* 2019. Vol. 8, N. 4. P. 287–291.
8. Kim H., Yoshimura N., Koike Y. Classification of Movement Intention Using Independent Components of Premovement EEG // *Frontiers in human neuroscience.* 2019. Vol. 13, N. 63.
9. Karakullukcu N., Yilmaz B. Detection of Movement Intention in EEG-Based Brain-Computer Interfaces Using Fourier-Based Synchrosqueezing Transform // *International journal of neural systems.* 2022. Vol. 32, N. 1.
10. Blankertz B., Müller K. R., Curio G., Vaughan T. M., Schalk G., Wolpaw J. R., Schlögl A., Neuper C., Pfurtscheller G., Hinterberger T., Schröder M., Birbaumer N. The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials // *IEEE Trans. Biomed. Eng.* 2004. Vol. 51.
11. Wang K., Xu M., Wang Y., Zhang S., Chen L., Ming D. Enhance decoding of pre-movement EEG patterns for braincomputer interfaces // *J. Neural Eng.* 2020. Vol. 17.
12. Quiles V., Ferrero L., Iáñez E., Ortiz M., Cano J. M., Azorín J. M. Detecting the Speed Change Intention from EEG Signals: From the Offline and Pseudo-Online Analysis to an Online Closed-Loop Validation // *Applied Sciences.* 2022. Vol. 12, N. 1.
13. Ibáñez J., Serrano, J. I., del Castillo M. D., Barrios L., Gallego J. A., Rocon E. An EEG-Based Design for the Online Detection of Movement Intention // *Lecture Notes in Computer Science.* Vol. 6691. P. 370–378.
14. Якубович В. А. Рекуррентные конечно-сходящиеся алгоритмы решения систем неравенств // *Доклады Академии наук СССР.* 1966. Т. 166, № 6. С. 1308–1312.
15. Липкович М. М., Миронов Д. В. Применение алгоритма "Полоска" в задаче онлайн-машинного обучения // *Интеллектуальные системы: теория и приложения.* 2021. Т. 25, № 4. С. 231–234.
16. Lipkovich M. Yakubovich's method of recursive objective inequalities in machine learning // *IFAC-PapersOnLine.* 2022. Vol. 55, N. 12. P. 138–143.
17. Якубович В. А. Три теоретические схемы обучаемых опознающих систем // *Самонастраивающиеся системы. Распознавание образов. Конечные автоматы и релейные устройства.* М.: Наука. 1967. С. 183–191.
18. Blankertz B., Curio G., Müller K. R. Classifying Single Trial EEG: Towards Brain Computer Interfacing // *Advances in Neural Inf. Proc. Systems 14 (NIPS 01).* 2002.
19. Ma J., Lawrence S., Stefan S., Voelker G. Identifying Suspicious URLs: An Application of Large-Scale Online Learning // *ICML09: Proceedings of the 26th Annual International Conference on Machine Learning.* 2009. P. 681688.

Application of the "Stripe" Algorithm for Online Decoding of the EEG Patterns

M. M. Lipkovich, lipkovich.mikhail@gmail.com,

Institute for Problems in Mechanical Engineering, Saint Petersburg, 199178, Russian Federation,

A. R. Sagatdinov, amazar226@gmail.com,

Saint Petersburg State University, Saint Petersburg, 198504, Russian Federation

Corresponding author: Lipkovich Mikhail M., Cand. of Phys. and Math.Sc., Senior Researcher, Institute for Problems in Mechanical Engineering of the Russian Academy of Sciences, Saint Petersburg, 199178, Russian Federation, e-mail: lipkovich.mikhail@gmail.com

Accepted on March 06, 2023

Abstract

In this paper, we consider the problem of determining the hand with which the subject intends to make a movement according to the signals of the electroencephalogram. The relevance of the task is due to the wide spread of brain-computer

interfaces, where electroencephalography is one of the main non-invasive methods for obtaining signals from the brain. To solve the problem, temporal and frequency features are selected from the segments of signals preceding the movement, which are fed to the input of the classification machine learning model. In contrast to the standard supervised learning setup, it is assumed that there is no predefined training data set and the training samples for the model are received one after another. Thus, a situation is simulated in which the model must work with a new subject and adjust to them in real time. The traditional method for training linear models in such a paradigm is stochastic gradient descent. Previously, it was shown that the "Stripe" algorithm developed by Yakubovich for a certain problem has a higher convergence rate than stochastic gradient descent. However, this is achieved by performing algorithm step on each feature of the sample. Thus, that version of "Stripe" is not suitable for working with high-dimensional data. This article discusses another version of "Stripe" that does not have this drawback. It is shown that the proposed algorithm has a higher rate of one learning step compared to traditional linear models based on stochastic gradient descent on the BCI competition II dataset.

Keywords: recurrent objective inequalities, brain-computer interfaces, EEG classification, machine learning

Acknowledgements: The research was supported by RSF (project No. 21-71-00144).

For citation:

Lipkovich M. M., Sagatdinov A. R. Application of the "Stripe" Algorithm for Online Decoding of the EEG Patterns, *Mekhatronika, Avtomatizatsiya, Upravlenie*, vol. 24, no. 6, pp. 300–306.

DOI: 10.17587/mau.24.300-306

References

1. Kawala-Sterniuk A., Browarska N., Al-Bakri A., Pelc M., Zygarlicki J., Sidikova M., Martinek R., Gorzelanczyk E. J. Summary of over Fifty Years with Brain-Computer Interfaces—A Review, *Brain Sciences*, 2021, vol. 11, no. 43.
2. Vilela M., Hochberg L. Applications of brain-computer interfaces to the control of robotic and prosthetic arms, *Handbook of Clinical Neurology*, 2020, vol. 168, pp. 87–99.
3. Bockbrader M., Francisco G., Lee R., Olson J., Solinsky R., Boninger M. Brain Computer Interfaces in Rehabilitation Medicine, *The Journal of Injury, Function, and Rehabilitation*, 2018, vol. 10.
4. Ruiz S., Birbaumer N., Sitaram R. Abnormal Neural Connectivity in Schizophrenia and fMRI-Brain-Computer Interface as a Potential Therapeutic Approach, *Frontiers in Psychiatry*, 2013, vol. 4.
5. Sebastián-Romagosa M., Cho W., Ortner R., Murovec N., Von Oertzen T., Kamada K., Allison B. Z., Guger C. Brain Computer Interface Treatment for Motor Rehabilitation of Upper Extremity of Stroke Patients—A Feasibility Study, *Frontiers in Neuroscience*, 2020, vol. 14.
6. McFarland D. J., Wolpaw J. R. EEG-Based Brain-Computer Interfaces, *Current Opinion in Biomedical Engineering*, 2017, vol. 4, pp. 194–200.
7. Plotnikov S. A., Lipkovich M., Semenov D. M., Fradkov A. L. Artificial intelligence based neurofeedback, *Cybernetics and Physics*, 2019, vol. 8, no. 4, pp. 287–291.
8. Kim H., Yoshimura N., Koike Y. Classification of Movement Intention Using Independent Components of Premovement EEG, *Frontiers in Human Neuroscience*, 2019, vol. 13, no. 63.
9. Karakullukcu N., Yilmaz B. Detection of Movement Intention in EEG-Based Brain-Computer Interfaces Using Fourier-Based Synchrosqueezing Transform, *International Journal of Neural Systems*, 2022, vol. 32, no. 1.
10. Blankertz B., Müller K. R., Curio G., Vaughan T. M., Schalk G., Wolpaw J. R., Schlögl A., Neuper C., Pfurtscheller G., Hinterberger T., Schröder M., Birbaumer N. The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials, *IEEE Transactions on Biomedical Engineering*, 2004, vol. 51.
11. Wang K., Xu M., Wang Y., Zhang S., Chen L., Ming D. Enhance decoding of pre-movement EEG patterns for brain-computer interfaces, *Journal of Neural Engineering*, 2020, vol. 17.
12. Quiles V., Ferrero L., Iáñez E., Ortiz M., Cano J. M., Azorín J. M. Detecting the Speed Change Intention from EEG Signals: From the Offline and Pseudo-Online Analysis to an Online Closed-Loop Validation, *Applied Sciences*, 2022, vol. 12, no. 1.
13. Ibáñez J., Serrano J. I., del Castillo M. D., Barrios L., Gallego J. Á., Rocon E. An EEG-Based Design for the Online Detection of Movement Intention, *Lecture Notes in Computer Science*, vol. 6691, pp. 370–378.
14. Yakubovich V. A. Recurrent finitely-convergent algorithms to solve the system of inequalities, *Doklady Akademii Nauk SSSR*, 1966, vol. 166, no. 6, pp. 1308–1312 (in Russian).
15. Lipkovich M. M., Mironov D. Application of "Stripe" algorithm for online machine learning, *Intelligent Systems. Theory and Applications*, 2021, vol. 25, no. 4, pp. 231–234 (in Russian).
16. Lipkovich M. Yakubovich's method of recursive objective inequalities in machine learning, *IFAC-PapersOnLine*, 2022, vol. 55, no. 12, pp. 138–143.
17. Yakubovich V. A. Three Theoretical Schemes of Learning Systems, in *Samoobuchayushchiesya Avtomaticheskije Sistemy*, Moscow, Nauka, 1967, pp. 183–191 (in Russian).
18. Blankertz B., Curio G., Müller K. R. Classifying Single Trial EEG: Towards Brain Computer Interfacing, *Advances in Neural Information Processing Systems 14 (NIPS 01)*, 2002, pp. 157–164.
19. Ma J., Lawrence S., Stefan S., Voelker G. Identifying Suspicious URLs: An Application of Large-Scale Online Learning, *ICML'09: Proceedings of the 26th Annual International Conference on Machine Learning*, 2009, pp. 681–688.