

Recenzja pracy doktorskiej magistra Piotra Tadeusza Różańskiego dla Wydziału Matematyki, Informatyki i Mechaniki Uniwersytetu Warszawskiego

Dr hab. Piotr Pokarowski

Recenzowanym dziełem Pana Różańskiego jest program komputerowy Enhanced Matching Pursuit Implementation (EMPI) opisany w publikacjach A-D, załączonych do rozprawy doktorskiej pt. „Heterogeniczna implementacja matching pursuit z symulacją ciągłej przestrzeni parametrów”, która z kolei stanowi podsumowanie wyników Autora.

Program EMPI implementuje algorytm Matching Pursuit (MP) obliczający rzadką aproksymację sygnału (zdyskretyzowanej funkcji lub wektora). Celem rzadkiej aproksymacji jest znalezienie dokładnego przybliżenia obserwowanego sygnału x za pomocą rzadkiej (krótkiej) sumy łatwych atomów (funkcji) – wybranych np. ze słownika (zbioru) Gabora, czyli unormowanych kosinusów pomnożonych przez gęstość rozkładu normalnego.

Algorytm MP inicjuje resztę wejściowym sygnałem: $r = x$. Następnie wykonuje w pętli dwie operacje:

1. wybiera ze słownika atom g o największym iloczynie skalarnym z r ,
2. uaktualnia resztę: $r = r - (r,g)g$.

Na koniec zwraca: $approx = x - r$.

Algorytm MP pojawił się w dziedzinie przetwarzania sygnału w 1993 r., w pracy [1], ale był znany już wcześniej - np w statystyce od 1960 r. pod nazwą „stepwise regression”. Nic dziwnego, jest to po prostu jeszcze jedna forma zachłannej heurystyki minimalizacyjnej typu forward lub bottom-up. Typowo liczność słownika jest większa od długości sygnału N , więc koszt jednej iteracji jest przynajmniej rzędu N^2 – zdecydowanie za dużo w niektórych zastosowaniach. Zrozumiałe zatem, że popularna implementacja MP, czyli Matching Pursuit ToolKit (MPTK) wykonuje iloczyny skalarne jednocześnie grupami, co w przypadku słownika Gabora jest równoważne splotowi sygnału, więc dzięki FFT i dodatkowym trickom koszt jednej iteracji MPTK jest rzędu $N \log(N)$.

1. EMPI nie jest istotną poprawą MPTK, natomiast FastMP jest istotną poprawą. Program MPTK został opublikowany w 2006 r. w pracy [2], a w 2013 r. – kiedy Pan Różański opublikował swoją pierwszą implementację MP (praca A) – MPTK miał 130 cytowań w bazie Google Scholar, był więc znany w środowisku zajmującym się rzadką aproksymacją. Niestety, Autor rozprawy nie porównuje swojej implementacji z MPTK: ani w pracy A z 2013, ani w pracy B z 2018 r., ani w pracy C z 2020. W tych pracach Autor w ogóle nie porównuje swojej implementacji z innymi! To niespotykana praktyka w pismach głównego nurtu obliczeń naukowych. Nie dziwię się więc, że prace A-C zostały opublikowane poza głównym nurtem.

W bieżącym roku ukazała się ostatnia z załączonych prac Pana Różańskiego na temat implementacji MP (praca D). Tym razem Autor porównuje EMPI, ale tylko z MPTK. W szczególności, z wykresów na Fig 6. wynika, że MPTK i EMPI, kiedy są wykonywane na jednym CPU, dają niemal identyczne wyniki. Ponadto w Tabeli 1. widzimy, że dzięki obsłudze GPU, program EMPI może być 4 razy szybszy od MPTK, co w podsumowaniu rozprawy Pan Różański przedstawia jako swój główny wynik: „zapropozowanie wysoce wydajnej implementacji”.

W pracy D wzmiankowany jest również inny program implementujący MP, nazwijmy go w skrócie FastMP, opublikowany w 2021 r. w pracy [3]. Pan Różański nie porównuje EMPI z FastMP, ponieważ – jego zdaniem – zaprogramowana w FastMP procedura zerowania małych elementów macierzy Grama „may introduce arbitrary errors into calculations”. Sprawdziłem w Tabeli 1. z pracy [3], że program FastMP jest szybszy od MPTK ok. 4 razy w małych zadaniach i ok. 70 razy w dużych, więc można się spodziewać, że FastMP jest o rząd wielkości szybszy od EMPI, choć nie obsługuje GPU.

Główną metodą poznawczą w obliczeniach naukowych są eksperymenty porównujące różne programy komputerowe w różnych benchmarkach, czyli zadaniach testowych, które powinny być reprezentatywne dla praktyki. Zaproponowanie nowego zadania to czasami najważniejszy wynik pracy. Jest dobrą praktyką nauk obliczeniowych, że gdy ostatnio opublikowano program Y i pokazano, że poprawia on w nowym zadaniu klasyczny program X, to autor nowej pracy o programie Z podaje wyniki działania X,Y,Z w tymże zadaniu (powtórzenie obliczeń dla X,Y też ma swoją wartość). Nie poprzestaje na stwierdzeniu, że Y może być gorszy od Z.

Na podstawie omówionych wyżej eksperymentów stwierdzam, że EMPI nie jest istotną poprawą MPTK, FastMP jest istotną poprawą, natomiast Pan Różański nie przedstawił porównania EMPI z FastMP. Wobec tego uważam, że prace A-D nie spełniają standardów obowiązujących w obliczeniach naukowych.

Na marginesie chciałbym wyrazić swoją opinię na temat faktu, że pismo ACM Trans. Math. Soft. – w którym ukazał się artykuł D – ma 200 punktów na liście MNIŚW. Uważam, że to zdecydowanie za dużo. Impact Factor sugeruje, że powinno być 100 lub 70. Owszem, ACM Trans. Math. Soft. jest zasłużonym pismem dla starszego pokolenia numeryków i jeszcze 25 lat temu wydawano w nim opisy ważnych procedur w Fortranie (np. praca Zhu et al. z 1997 r. o implementacji L-BFGS-B – ok. 4 tys. cyt.). Jednak obecnie, prestiżowe pisma drukujące głównie raporty o nowym oprogramowaniu mają Impact Factor większy od 10 (np. J. Stat. Soft., Nucleic Acids Research, nie wspominając o najważniejszych konferencjach AI, które IF formalnie nie mają, ale można go estymować). Tymczasem ACM Trans. Math. Soft. ma Impact Factor równy 2.

2. Autor nie uzasadnił, że słowniki używane przez EMPI są optymalne w sensie rzadkiej aproksymacji, którą oblicza MP. Program EMPI używa słowników, w których każde 2 najbliższe atomy są odległe o stałą wartość. W aproksymacji pisze się o takich zbiorach, że są jednakowo rozłożone, natomiast Autor używa życzeniowego nazewnictwa: „optymalny słownik” „optymalna konstrukcja słownika”, „optymalna struktura słownika” – wszystko bez wyjaśnienia związków tych obiektów z celem obliczeń MP. Przy czym wyjaśnienie mogłoby być teoretyczne lub eksperymentalne.

Z punktu widzenia teorii, formalnym celem rzadkiej aproksymacji jest minimalizacja błędu l^2 przy ograniczeniu na liczbę atomów w aproksymacji lub minimalizacja liczby atomów w aproksymacji przy ograniczeniu na błąd l^2 . Aby zatem pokazać przewagę słowników jednakowo rozłożonych można by np. udowodnić twierdzenie mówiące, że takie słowniki zmniejszają liczbę atomów w aproksymacji. Z punktu widzenia eksperymentów, można by pokazać w realistycznych benchmarkach, że EMPI istotnie zmniejsza błąd aproksymacji przy ustalonej liczbie atomów w aproksymacji w stosunku do MPTK.

Autor nie podał wyników teoretycznych na temat optymalności słowników jednakowo rozłożonych, natomiast prawy wykres na Fig. 6. z pracy D wskazuje, że rezultat obliczeń za pomocą EMPI i MPTK jest identyczny: „the resulting decomposition and the corresponding residual signal energy is exactly the same for both implementations.” Tak więc rozważania dotyczące optymalności uważam za nieistotne.

3. MP nie odgrywa istotnej roli ani dla rzadkiej aproksymacji ani dla analizy sygnału EEG.

Pan Różański wyjaśnia w rozprawie, że pracował nad EMPI w związku z badaniami prowadzonymi na Wydziale Fizyki UW na temat przetwarzania sygnałów biomedycznych, głównie EEG. Ponadto kilkakrotnie, ale bez uzasadnienia, stwierdza, że MP „odgrywa istotną rolę” w analizie sygnału EEG. W ogóle, w rozprawie brak szerszego kontekstu prowadzonych badań: choćby wspomnienia, że w rzadkiej aproksymacji są jeszcze inne, nowoczesne algorytmy oprócz MP lub uwagi, że w przetwarzaniu sygnału EEG, algorytm MP służy do odsumiania czy też usuwania artefaktów. Aby zorientować się, czy Autor ma prawidłowe wyobrażenie o dziedzinie, w której pracuje, przeczytałem kilka nowych prac przeglądowych oraz wykonałem wstępną analizę bibliometryczną. Na takiej podstawie stwierdzam, że algorytm MP i w konsekwencji badania Autora, nie mają istotnego znaczenia:

1. ani w dziedzinie rzadkiej aproksymacji,
2. ani w dziedzinie analizy sygnału EEG.

Ad 1. Kilka lat po wprowadzeniu MP, pojawiły się niezależnie, w różnych kontekstach, algorytmy rozwiązujące zadanie rzadkiej regularyzacji (RR), czyli zadanie najmniejszych kwadratów z karą l^1 – np. w statystyce: Lasso [4], w analizie sygnału: Basis Pursuit Denoising (BPDN) [5]. Oryginalne implementacje nie były efektywne, dlatego na przełomie wieków MP i MPTK miały znaczenie dla nauki. Jednak ok. 2010 r. powstały efektywne implementacje RR [6-8] i od tego czasu zainteresowanie algorytmami MP i MPTK maleje. Gdyby porównać MPTK i jedną z tych implementacji RR za pomocą liczby cytowań w Google Scholar, otrzymalibyśmy iloraz rzędu 1/100. To może być rozsądna miara względnego znaczenia metod MP i RR dla rzadkiej aproksymacji w ostatnich 20 latach.

Ad 2. W analizie sygnału EEG, algorytmy rzadkiej aproksymacji są używane do odsumiania czy też usuwania artefaktów pomiarowych obok metod regresyjnych, blind source separation, transformacji falkowej, empirical mode decomposition, filtrów adaptacyjnych czy innych metod. Według przeglądu metod odsumiania sygnału EEG z 2019 r. [9], znaczenie metod rzadkiej aproksymacji jest niewielkie, dominuje independent component analysis (ICA). W przeglądówce na temat przetwarzania sygnału EEG z 2023 r. [10], nie ma już mowy o rzadkiej aproksymacji, za to potwierdza się znaczenie ICA. Z kolei, jeśli zamiast polegać na opiniach z przeglądówek, porównamy:

- cytowania implementacji MP – np. rozwijanego przez prof. Durkę programu Svarog (praca A)
- z implementacją ICA – np. rozwijanego przez prof. Makeiga programu EEGLAB [11],

to otrzymamy iloraz rzędu 1/1000.

Biorąc pod uwagę, że często odsumianie to tylko jeden z etapów przetwarzania sygnału EEG, którego celem jest klasyfikacja pracy mózgu, warto na koniec porównać najbardziej cytowane klasyfikatory. I tak, powstały w 2019 r. w labie prof. Makeiga, klasyfikator ICLabel oparty na ICA [12] był cytowany 1,4 tys razy, natomiast powstała niemal w tym samym czasie głęboka sieć konwolucyjna EEGNet [13] była cytowana 3,1 tys. razy. To może wyjaśnia, dlaczego twórcy udanego programu MPTK pracują dzisiaj w AI.

4. Konkluzja. Uważam, że rozprawa doktorska mgr Piotra Tadeusza Różańskiego nie spełnia warunków określonych w ustawie „Prawo o szkolnictwie wyższym i nauce” i nie jest zdalna do publicznej obrony.

Piotr Pokarowski

Prace cytowane

[1] Mallat, S. G., & Zhang, Z. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on signal processing*, 41(12), 3397-3415. (13 tys cyt.)

- [2] Krstulovic, S., & Gribonval, R. (2006). MPTK: Matching pursuit made tractable. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*. **(206 cyt.)**
- [3] Průša, Z., Holighaus, N., & Balazs, P. (2021). Fast matching pursuit with multi-Gabor dictionaries. *ACM Transactions on Mathematical Software (TOMS)*, 47(3), 1-20. **(7 cyt.)**
- [4] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1), 267-288. **(61 tys. cyt.)**
- [5] Chen, S. S., Donoho, D. L., & Saunders, M. A. (2001). Atomic decomposition by basis pursuit. *SIAM review*, 43(1), 129-159. **(15 tys. cyt.)**
- [6] Beck, A., & Teboulle, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1), 183-202. **(14 tys. cyt.)**
- [7] Friedman, J. H., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33, 1-22. **(19 tys. cyt.)**
- [8] Boyd, S., Parikh, N., Chu, E., Peleato, B., & Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1), 1-122. **(24 tys. cyt.)**
- [9] Jiang, X., Bian, G. B., & Tian, Z. (2019). Removal of artifacts from EEG signals: a review. *Sensors*, 19(5), 987. **(765 cyt.)**
- [10] Chaddad, A., Wu, Y., Kateb, R., & Bouridane, A. (2023). Electroencephalography signal processing: A comprehensive review and analysis of methods and techniques. *Sensors*, 23(14), 6434. **(74 cyt.)**
- [11] Delorme, A., & Makeig, S. (2004). EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of neuroscience methods*, 134(1), 9-21. **(24 tys. cyt.)**
- [12] Pion-Tonachini, L., Kreutz-Delgado, K., & Makeig, S. (2019). ICLLabel: An automated electroencephalographic independent component classifier, dataset, and website. *NeuroImage*, 198, 181-197. **(1,4 tys. cyt.)**
- [13] Lawhern, V. J., Solon, A. J., Waytowich, N. R., Gordon, S. M., Hung, C. P., & Lance, B. J. (2018). EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces. *Journal of neural engineering*, 15(5), 056013. **(3,1 tys. cyt.)**