ComNetX: локальная иерархическая адаптация методов обнаружения сообществ в динамических графах

Г. В. Боков*, А. Ю. Коновалов‡, А. С. Упорова§, А. С. Дробышев¶, Я. Г. Егоров‖

Представлен алгоритм ComNetX, адаптирующий статические методы обнаружения сообществ к динамическим графам посредством локального обновления иерархии разбиений. Поддерживаются топологические и нейросетевые методы, в том числе использующие признаки вершин. Установлены условия сохранения вложенности разбиений и модулярности; получены оценки времени и памяти. На пяти графах ускорение относительно повторного запуска Leiden составляет 6,60–98,72 раза при абсолютной разности итоговых модулярностей 0,0035–0,0213.

Ключевые слова: динамический граф, обнаружение сообществ, иерархия разбиений, модулярность.

Постановка задачи. Задача обнаружения сообществ заключается в разбиении вершин графа на группы с более высокой плотностью связей внутри групп, чем между ними. В динамическом графе разбиение необходимо обновлять после каждого пакета изменений ребер. Требуется сократить вычислительные затраты по сравнению с полным пересчетом разбиения без существенной потери качества.

Пусть $V = \{1, \ldots, n\}$ — фиксированное множество вершин, $A_t \in \mathbb{R}^{n\times n}_{\geq 0}$ — матрица смежности после t-го пакета изменений, $\Delta_t \in \mathbb{R}^{n\times n}$ —

* *Боков Григорий Владимирович* — Исследовательский центр в сфере ИИ МГУ, e-mail: grigoriy.bokov@math.msu.ru.

Bokov Grigoriy — AI Research Center, Lomonosov Moscow State University.

‡ *Коновалов Александр Юрьевич* — Исследовательский центр в сфере ИИ МГУ, e-mail: alexandr.konoval@gmail.com.

Konovalov Aleksandr — AI Research Center, Lomonosov Moscow State University.

§ *Упорова Анна Сергеевна* — Исследовательский центр в сфере ИИ МГУ, e-mail: annuporova2003@gmail.com.

Uporova Anna — AI Research Center, Lomonosov Moscow State University.

¶ *Дробышев Александр Сергеевич* — Исследовательский центр в сфере ИИ МГУ, e-mail: drobyshev.sanya@yandex.ru.

Drobyshev Alexander — AI Research Center, Lomonosov Moscow State University.

‖ *Егоров Ярослав Григорьевич* — Исследовательский центр в сфере ИИ МГУ, e-mail: egorovyg@my.msu.ru.

Egorov Iaroslav — AI Research Center, Lomonosov Moscow State University.

Опубликовано на условиях лицензии CC-BY 4.0
Published under CC-BY 4.0

матрица приращений, тогда $A_t = A_{t-1} + \Delta_t$. Допускаются отрицательные приращения при неотрицательности итоговых весов. Признаки вершин, если они используются, задаются неизменяемой матрицей $X \in \mathbb{R}^{n \times d}$, где d — число признаков.

Качество разбиения оценивается модулярностью [1]. Для графа на h вершинах с матрицей смежности $A \in \mathbb{R}_{\geq 0}^{h \times h}$ и метками сообществ $c \in \{1, \ldots, h\}^h$ она задается формулой:

$$Q_\gamma(A, c) = \frac{1}{m} \sum_{i,j=1}^{h} \left(A_{ij} - \gamma \frac{k_i^{\text{out}} k_j^{\text{in}}}{m} \right) \mathbf{1}\{c_i = c_j\}, \tag{1}$$

где $m = \sum_{i,j} A_{ij} > 0$, $k_i^{\text{out}} = \sum_j A_{ij}$, $k_j^{\text{in}} = \sum_i A_{ij}$ — исходящие и входящие взвешенные степени, $\gamma > 0$ — параметр резолюции, $\mathbf{1}\{\cdot\}$ — индикатор.

Алгоритм ComNetX. ComNetX вызывает статический метод $\mathcal{B}$ на сжатом подграфе и хранит $L \in \mathbb{N}$ разбиений $\mathcal{P}_t^\ell$, где t — номер пакета обновлений, $\ell = 0, \ldots, L-1$ — уровень. Результатом служит разбиение $\mathcal{P}_t^0$. Leiden [2] строит начальное разбиение $\mathcal{P}_0^0$. Для получения каждого из следующих $L-1$ уровней он применяется к графу сообществ предыдущего уровня, после чего найденные метки переносятся на V. Метки $C_t^\ell \in \mathbb{Z}^n$ равны минимальным номерам вершин сообществ: $\mathcal{P}_t^\ell = \big\{\{v \in V : C_t^\ell(v) = a\} : a \in \mathbb{Z}\big\} \setminus \{\varnothing\}$. Пакет обновлений обрабатывается в 4 этапа:

1. Выбор областей. Множества $S_t = \bigcup_{i,j \in V : (\Delta_t)_{ij} \neq 0} \{i, j\}$ и $B_t = \{v \in V : \operatorname{dist}(v, S_t) \leq r\}$ задают концы измененных ребер и их окрестность радиуса $r \in \mathbb{Z}_{\geq 0}$ в A_t без учета весов и направлений. Области пересчета $U_t^\ell = \bigcup_{H \in \mathcal{P}_{t-1}^\ell : H \cap B_t \neq \varnothing} H$ фиксируются на весь обрабатываемый пакет; этапы 2–4 выполняются от уровня 0 к $L-1$.

2. Выделение атомов сжатия. Атомами называются непустые множества вершин, сжимаемые в одну вершину. При $\ell = 0$ строится $\mathcal{G}_t^0 = \big\{\{v\} : v \in B_t\big\} \cup \big\{(H \cap U_t^0) \setminus B_t : H \in \mathcal{P}_{t-1}^{L-1}\big\} \setminus \{\varnothing\}$, причем вершины B_t не сжимаются. При $\ell > 0$ обновленные сообщества сжимаются: $\mathcal{G}_t^\ell = \{H \in \mathcal{P}_t^{\ell-1} : H \subseteq U_t^\ell\}$.

3. Построение сжатого графа. Пусть $\mathcal{G}_t^\ell = \{G_1^\ell, \ldots, G_{k_\ell}^\ell\}$, $k_\ell = |\mathcal{G}_t^\ell|$, а $P_\ell \in \{0,1\}^{k_\ell \times n}$ — матрица принадлежности, $(P_\ell)_{ai} = \mathbf{1}\{i \in G_a^\ell\}$. Матрица смежности $\bar{A}_\ell = P_\ell A_t P_\ell^\top \in \mathbb{R}_{\geq 0}^{k_\ell \times k_\ell}$ и, при наличии X, матрица признаков $\bar{X}_\ell \in \mathbb{R}^{k_\ell \times d}$ задают сжатый граф на уровне ℓ. Внутренние ребра образуют петли, а признаки вершин усредняются по атомам.

4. Пересчет и перенос меток. Метод $\mathcal{B}$ находит сообщества в сжатом графе. Все исходные вершины атомов одного сообщества получают метку, равную минимальному номеру среди этих вершин. Вне области пересчета U_t^ℓ метки сохраняются.

Теоретические свойства. Сохраняется вложенность $\mathcal{P}_t^0 \preceq \cdots \preceq \mathcal{P}_t^{L-1}$, поскольку области удовлетворяют $U_t^{\ell-1} \subseteq U_t^{\ell}$ при $\ell > 0$, обновленные сообщества сжимаются целиком и не разделяются на следующем уровне, а вне областей пересчета метки не меняются. Подграф на U_t^{ℓ} содержит все ребра между его вершинами. Сохранение внутренних ребер атомов в виде петель обеспечивает неизменность суммарного веса, а входящие и исходящие степени суммируются по атомам. Поэтому при положительном суммарном весе и одинаковом γ из формулы (1) следует равенство модулярностей разбиения сжатого графа и его переноса на U_t^{ℓ}.

Оценки времени и памяти. Для заданной A_t время обработки пакета без учета вызовов $\mathcal{B}$ составляет $O(r(n+s_t)+L[(n+s_t)\log(2+n+s_t)+F])$, где s_t — число ненулевых элементов A_t, $F = nd$ при наличии X и 0 иначе. Для хранения графа, меток и признаков требуется $O(s_t + Ln + F)$ памяти.

Экспериментальные результаты. Полный пересчет и ComNetX сравнивались с Leiden на графах Cora, ACM, CiteSeer, PubMed и ArxivMath[1] и с S^{2}CAG [3] на графах PubMed и ArxivMath ($L = 3$, $r = 1$, $\gamma = 1$)[2].

Сначала загружался граф из 99,9% перемешанных записей ребер, затем оставшиеся 0,1% подавались обоим методам за 5–10 непустых последовательных пакетов обновлений примерно равного размера.

Ускорение вычислялось как отношение суммарных времен полного пересчета и ComNetX без учета времени, затраченного на добавление ребер, преобразование форматов и оценку качества. Итоговая модулярность вычислялась после обработки последнего пакета. Leiden запускался на всем графе с предыдущим разбиением.

	Итоговая модулярность Leiden		Ускорение, раз		
Граф	Полный	ComNetX	Leiden	S^{2}CAG, 5 пак.	S^{2}CAG, 10 пак.
Cora	0,8240	0,8141	6,60	—	—
ACM	0,8255	0,8042	7,54	—	—
CiteSeer	0,8932	0,8897	8,43	—	—
PubMed	0,7872	0,7822	30,12	3,44	3,37
ArxivMath	0,8858	0,8812	98,72	37,63	70,40

Leiden обрабатывал 8 непустых пакетов на CiteSeer и 10 на остальных графах. Для S^{2}CAG указана медиана ускорений пяти пар запусков с фиксированными случайными признаками: $X_{ij} \sim N(0,1)$, $d = 64$, $seed = 42$.

Благодарности. Работа поддержана Минэкономразвития России (соглашение № 139-15-2025-012, идентификатор 000000C313925P4H0002). Вычисления выполнены на суперкомпьютере МГУ-270.

[1] Графы взяты из следующих библиотек: Cora, ACM, CiteSeer — Awesome Deep Graph Clustering; PubMed — PyTorch Geometric; ArxivMath — Data4TGC.

[2] Реализация: `https://github.com/mpailab/comnetx`.

Список литературы

[1] M. E. J. Newman, M. Girvan, “Finding and evaluating community structure in networks”, *Physical Review E*, **69**:2 (2004), 026113. DOI: 10.1103/PhysRevE.69.026113.

[2] V. A. Traag, L. Waltman, N. J. van Eck, “From Louvain to Leiden: guaranteeing well-connected communities”, *Scientific Reports*, **9** (2019), 5233. DOI: 10.1038/s41598-019-41695-z.

[3] X. Lin, R. Yang, H. Zheng, X. Ke, “Spectral Subspace Clustering for Attributed Graphs”, *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, **1**, ACM, 2025, 789–799. DOI: 10.1145/3690624.3709207 .

Статья поступила 10 ноября 2026 г.

ComNetX: Local Hierarchical Adaptation for Batched Dynamic Community Detection

G. Bokov, A. Konovalov, A. Uporova, A. Drobyshev, I. Egorov

We present ComNetX, an algorithm that adapts static community detection methods to dynamic graphs through local updates of a partition hierarchy. It supports topology-based and neural methods, including those using node features. We establish preservation of hierarchy nesting, conditions for modularity preservation under graph contraction, and time and memory bounds. On five graphs, speedups over repeated Leiden runs range from 6.60 to 98.72, with absolute final modularity differences of 0.0035–0.0213.

Keywords: dynamic graph, community detection, partition hierarchy, modularity.

References

[1] M. E. J. Newman, M. Girvan, “Finding and evaluating community structure in networks”, *Physical Review E*, **69**:2 (2004), 026113. DOI: 10.1103/PhysRevE.69.026113.

[2] V. A. Traag, L. Waltman, N. J. van Eck, “From Louvain to Leiden: guaranteeing well-connected communities”, *Scientific Reports*, **9** (2019), 5233. DOI: 10.1038/s41598-019-41695-z.

[3] X. Lin, R. Yang, H. Zheng, X. Ke, “Spectral Subspace Clustering for Attributed Graphs”, *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, **1**, ACM, 2025, 789–799. DOI: 10.1145/3690624.3709207.

Received on November 10, 2026