АНОТАЦІЯ

Зважій Д. В. Індексний доступ на основі суфіксних масивів для підрядкового пошуку в PostgreSQL. — Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 122 «Комп'ютерні науки» (12 — Інформаційні технології). — Національний університет «Києво-Могилянська академія», Київ, 2026.

Дисертацію присвячено розробці методу індексного доступу для підрядкового пошуку, безпосередньо вбудованого в реляційну систему керування базами даних (СКБД). Підрядковий пошук, тобто знаходження усіх рядків бази даних, що містять заданий шаблон як префікс, як суфікс або як довільне входження, є фундаментальною операцією інформаційно-пошукових систем, проте у наявних реляційних СКБД він підтримується лише фрагментарно: вбудовані механізми PostgreSQL (Б-дерево з `text_pattern_ops`, GIN із `pg_trgm`, повнотекстовий пошук) охоплюють лише часткові випадки задачі, а зовнішні рішення на основі суфіксного індексу (зокрема SAES над Elasticsearch) не дають внутрішньої підтримки в самій реляційній СКБД. Це робить актуальною розробку методу індексного доступу, безпосередньо вбудованого в реляційну СКБД, який єдиною індексною структурою підтримує усі три операції підрядкового пошуку, визначений у термінах інваріантів цієї платформи та апробований у PostgreSQL.

Об'єктом дослідження є процес виконання операцій підрядкового пошуку в реляційних системах керування базами даних. **Предметом дослідження** є моделі, методи та програмні засоби індексного доступу на основі суфіксних структур даних, що підтримують виконання операцій підрядкового пошуку у середовищі реляційної СКБД-платформи. **Метою дослідження** є побудова теоретичної моделі, розробка та експериментальне обґрунтування методу індексного доступу на основі суфіксного масиву для підтримки операцій підрядкового пошуку в PostgreSQL. Метод має гарантувати логарифмічну складність пошуку $O(m \log N)$, однакову для трьох операцій (за префіксом, суфіксом та входженням), повну інтеграцію з менеджером буферів, журналюванням із записом наперед та програмним інтерфейсом методів доступу PostgreSQL і коректну взаємодію з моделлю багатоверсійного керування паралельністю на шляху читання.

Для досягнення мети поставлено такі **завдання** дослідження: формально визначити операції підрядкового пошуку та їхню ієрархію складності; проаналізу-

вати суфіксні структури даних як теоретичну основу методу; обґрунтувати вибір PostgreSQL як платформи апробації методу; виміряти продуктивність наявних механізмів PostgreSQL та оцінити розрив з теоретичними можливостями; побудувати теоретичну модель сторінкового суфіксного масиву у моделі блочного зовнішнього зберігання та довести межі I/O- і просторової складності методу інваріантно щодо конкретної СКБД-платформи; проаналізувати архітектуру PostgreSQL як систему обмежень для проєктування методу; обґрунтувати вибір суфіксного масиву порівняно з суфіксним деревом з урахуванням цих обмежень; розробити структуру сторінок, алгоритми побудови та пошуку методу; розробити програмну реалізацію SAindex безпосередньо в ядрі PostgreSQL; провести порівняльне експериментальне оцінювання на реальних та синтетичних даних.

Для вирішення цих завдань застосовано теоретичні методи аналізу алгоритмів у моделі блочного зовнішнього зберігання з параметрами $\langle B, e, e_k, N\rangle$ (побудова теоретичної моделі сторінкового суфіксного масиву та доведення меж I/O-складності методу); методи системної інтеграції через інтерфейс Access Method API PostgreSQL та системного програмування мовою C (розробка програмної реалізації SAindex); непараметричні методи статистичного аналізу, а саме критерій Манна–Вітні з контролем розміру ефекту через дельту Кліффа, якими виконано емпіричну верифікацію переваги на трьох наборах даних (GeoNames, synthetic_uniform, synthetic_zipf) на п’яти масштабах від $100\,000$ до $5\,000\,000$ рядків, по 30 запусків на шаблон.

Наукова новизна одержаних результатів полягає в такому.

Уперше розроблено теоретичну модель сторінкового суфіксного масиву, сумісного з інваріантами реляційної СКБД-платформи, у системі параметрів $\langle B, e, e_k, N\rangle$, яка зводить три операції підрядкового пошуку (префіксну, суфіксну та за входженням) до спільного діапазонного примітива з позиційною фільтрацією та обґрунтовує розподіл даних за схемою доступу через лему про розділення.

Уперше доведено межі I/O-складності методу: верхню межу пошукової фази $\leq\ 2\lceil\log_2 \mathcal{P}\rceil + O(1)$ блочних доступів незалежно від типу операції та довжини шаблону, лінійні нижні межі фази збирання результатів для трьох стратегій та повноту покриття множин результатів, інваріантні щодо конкретної реляційної СКБД-платформи.

Уперше встановлено просторову складність методу як $N(e + 8{,}125) + B$ бай-

тів, строго лінійну за обсягом колекції та незалежну від характеру даних, з кількісним обґрунтуванням коефіцієнта просторової надлишковості відносно вихідних даних.

Уперше запропоновано метод індексного доступу на основі сторінкового суфіксного масиву, повний цикл функціонування якого (зберігання, побудова, пошук, збирання результатів) специфіковано не в термінах PostgreSQL, а в термінах загальних інваріантів реляційної СКБД (буферний менеджер, журналювання із записом наперед, модель багатоверсійного керування паралельністю, інтерфейс методів доступу). Така платформо-незалежна специфікація уможливлює перенесення методу на будь-яку реляційну СКБД, що задовольняє ці інваріанти.

Дістала подальшого розвитку інтеграція суфіксних структур у систему типів та операторів реляційної СКБД через клас операторів `text_sa_ops` та два нові оператори: `@>` для пошуку за входженням (раніше доступний у PostgreSQL лише для контейнерних типів, але не для пари `(text, text)`) і `~@` для суфіксного пошуку, що є повністю новим оператором у системному каталозі; префіксний пошук інтегровано з вбудованим `^@`. На відміну від евристичного розпізнавання планувальником шаблонів `LIKE`, запропоноване розширення дозволяє виразити тип підрядкового запиту прямо у системному каталозі.

Практичне значення одержаних результатів та апробація методу. Запропонований метод апробовано шляхом розробки програмної реалізації SAindex обсягом 2782 рядки коду мовою C, інтегрованої безпосередньо в ядро PostgreSQL 18devel з відкритим вихідним кодом. Реалізація задовольняє вимоги стандартного протоколу побудови індексів та виконує повне WAL-журналювання операцій модифікації індексних сторінок; повноцінна підтримка інкрементальних модифікацій (`aminsert`, `ambulkdelete`, `amvacuumcleanup`) є напрямком подальшої роботи. На основі непараметричного критерію Манна–Вітні з контролем розміру ефекту через дельту Кліффа встановлено перевагу SAindex над триграмним методом `GIN_TRGM` з прискоренням 6,3$\times$ у медіані за шаблонами для запитів за входженням та 46,5$\times$ для префіксних запитів, перевагу над спеціалізованими Б-деревами у режимах низької селективності та рівнозначну продуктивність з `BTREE_REV` у режимі автоматичного вибору плану оптимізатором на масштабі 5M рядків. Достовірність експериментальних результатів підтверджено обов’язковою перевіркою коректності (160 з 160 тестових випадків відповідності множин рядків, повернутих SAindex, результатам послідовного

сканування) та статистичною значущістю на основі 2188 тестів попарного порівняння методів.

Дисертаційна робота складається з п'яти розділів.

У першому розділі проведено аналітичний огляд методів пошуку підрядків та теоретичних основ суфіксних структур даних. Формалізовано три цільові операції підрядкового пошуку (префіксний, суфіксний пошук та пошук за входженням); розглянуто фундаментальні суфіксні структури (суфіксне дерево, суфіксний масив, масив LCP, FM-індекс) та сучасні алгоритми їхньої побудови. Проаналізовано наявні механізми підрядкового пошуку в PostgreSQL та зовнішні рішення на основі суфіксного індексу; виявлено принциповий розрив між теоретичними можливостями суфіксних структур та їхньою реалізацією в реляційних СКБД, що обґрунтувало задачу дослідження.

У другому розділі виконано експериментальну оцінку наявних механізмів підрядкового пошуку: міжсистемне порівняння PostgreSQL, Elasticsearch, MySQL та ClickHouse з еталонною реалізацією суфіксного дерева, деталізований бенчмарк механізмів PostgreSQL з виявленням характеру їхніх обмежень. Побудовано спрощену модель вартості пошуку у сторінковому суфіксному масиві в параметрах $\langle B, e, N, L, \mathcal{P} \rangle$ та кількісно оцінено очікувану перевагу.

У третьому розділі побудовано теоретичну модель індексного доступу на основі суфіксного масиву у зовнішній пам'яті в параметрах $\langle B, e, e_k, N \rangle$, незалежну від конкретної СКБД-платформи. Доведено зведення трьох операцій підрядкового пошуку до єдиного діапазонного примітива з позиційною фільтрацією, межі I/O-складності пошукової та збиральної фаз, повноту покриття множин результатів і просторову складність $N(e + 8{,}125) + B$ байтів.

У четвертому розділі викладено метод індексного доступу на основі сторінкового суфіксного масиву для PostgreSQL. Архітектуру PostgreSQL подано як систему шести інваріантів C1–C6, що формують простір допустимих рішень; систематично оцінено шість вбудованих методів індексного доступу і обґрунтовано непридатність кожного як носія суфіксної структури. На цій основі запропоновано п'ятисекційну структуру зберігання на диску, формат індексного запису з матеріалізованим префіксом ключа, п'ятифазний конвеєр побудови через `tuplesort`, клас операторів `text_sa_ops` з трьома стратегіями пошуку та три стратегії збирання ідентифікаторів кортежів. Доведено коректність методу через систему з п'яти формальних тверджень.

У п’ятому розділі наведено опис програмної реалізації SAindex та її експериментальної оцінки. Реалізацію виконано мовою C обсягом 2782 рядки коду й інтегровано в ядро PostgreSQL 18devel. Сформульовано методологію оцінки на трьох наборах даних (GeoNames, synthetic_uniform, synthetic_zipf) на п’яти масштабах від 100 000 до 5 000 000 рядків, по 30 запусків на шаблон. Наведено результати порівняння SAindex з `GIN_TRGM`, `BTREE` та `BTREE_REV` за чотирма парами, проаналізовано їхню статистичну значущість, поведінку за обмеженого I/O та валідність вибору планувальника; обговорено практичні обмеження методу та намічено напрями подальшої роботи.

Ключові слова: суфіксний масив, дерева, Б-дерево, підрядковий пошук, обробка природної мови, метод індексного доступу, модель зовнішньої пам’яті, PostgreSQL, попарні порівняння, булева алгебра, прийняття рішень, оцінювання, архітектура системи, комп’ютерна система, інформаційна технологія.

ABSTRACT

Zvazhii D. V. Suffix-array-based index access for substring search in PostgreSQL. — Qualification scientific work submitted as a manuscript.

PhD thesis for the degree of Doctor of Philosophy in speciality 122 “Computer Science” (field of knowledge 12 “Information Technologies”). — National University of Kyiv-Mohyla Academy, Kyiv, 2026.

The dissertation is devoted to the development of an index access method for substring search natively integrated into a relational database management system. Substring search — locating all database strings that contain a given pattern as a prefix, as a suffix, or as an arbitrary occurrence — is a foundational operation of information-retrieval systems, yet existing relational DBMSs support it only fragmentarily: the built-in PostgreSQL mechanisms (the B-tree with `text_pattern_ops`, GIN with `pg_trgm`, full-text search) cover only partial cases of the problem, and external solutions based on a suffix index (notably SAES over Elasticsearch) do not provide in-DBMS support inside the relational system itself. This motivates the development of an index access method, natively integrated into a relational DBMS, that supports all three substring operations through a single index structure, is specified in terms of the invariants of that platform, and is validated on PostgreSQL.

The object of the research is the process of executing substring-search operations in relational database management systems. **The subject of the research** is the models, methods, and software facilities of index access based on suffix data structures that support the execution of substring-search operations within a relational DBMS platform. **The aim of the research** is to construct a theoretical model, design, and experimentally substantiate a suffix-array-based index access method supporting substring-search operations in PostgreSQL. The method is required to deliver logarithmic search complexity $O(m \log N)$, uniform across the three operations (prefix, suffix, and contains), full integration with the buffer manager, write-ahead log, and access-method interface of PostgreSQL, and correct interoperation with multi-version concurrency control on the read path.

To attain this aim, the following **research tasks** have been formulated: to formally define substring-search operations and their complexity hierarchy; to analyse suffix data structures as the theoretical foundation of the method; to justify the choice of PostgreSQL as the platform for validation of the method; to measure the performance

of existing PostgreSQL mechanisms and to assess the gap to the theoretical capabilities; to construct a theoretical model of a paged suffix array in the external-memory block model and prove the I/O and space complexity bounds of the method invariantly with respect to the particular relational DBMS platform; to analyse the PostgreSQL architecture as a system of constraints for the design of the method; to justify the choice of a suffix array versus a suffix tree under those constraints; to design the on-page structure and the build and search algorithms of the method; to develop a software implementation of SAindex integrated directly into the PostgreSQL core; and to carry out a comparative experimental evaluation on real and synthetic data.

To address these tasks, the work applies theoretical methods of algorithm analysis in the external-memory block model with parameters $\langle B, e, e_k, N\rangle$ (derivation of the theoretical model of the page-based suffix array and proof of the I/O complexity bounds of the method); methods of systems integration through the PostgreSQL Access Method API and of systems programming in C (development of the SAindex software implementation); and non-parametric methods of statistical analysis, specifically the Mann–Whitney U test combined with effect-size control via Cliff's delta, by which the advantage is empirically verified on three datasets (GeoNames, synthetic_uniform, synthetic_zipf) at five dataset sizes from $100{,}000$ to $5{,}000{,}000$ rows with 30 runs per pattern.

Scientific novelty of the obtained results.

For the first time, a formal model of a paged suffix array compatible with the invariants of a relational DBMS platform has been developed within the parameter system $\langle B, e, e_k, N\rangle$, which reduces the three substring-search operations (prefix, suffix, and contains) to a single range-search primitive with positional filtering and substantiates the access-pattern-based data layout through a separation lemma.

For the first time, the I/O complexity bounds of the method have been proven: an upper bound on the search phase of $\leq 2\lceil \log_2 \mathcal{P}\rceil + O(1)$ block accesses regardless of the operation type and pattern length, linear lower bounds on the result-collection phase for the three strategies, and the completeness of result-set coverage, invariant with respect to the particular relational DBMS platform.

For the first time, the space complexity of the method has been established as $N(e + 8.125) + B$ bytes, strictly linear in the size of the collection and independent of the nature of the data, with a quantitative justification of the space-amplification coefficient relative to the raw data.

For the first time, an index access method based on a paged suffix array is proposed, whose complete lifecycle (storage, build, search, result collection) is specified not in terms of PostgreSQL but in terms of the general invariants of a relational DBMS (buffer manager, write-ahead log, multi-version concurrency control, access-method interface). This platform-independent specification makes the method transferable to any relational DBMS that satisfies these invariants.

Further developed is the integration of suffix structures into the type-and-operator system of a relational DBMS through the `text_sa_ops` operator class and two new operators: `@>` for contains search (previously available in PostgreSQL only for container types, but not for the pair `(text, text)`) and `~@` for suffix search, which is an entirely new operator in the system catalogue; prefix search is integrated with the existing built-in `^@`. Unlike the heuristic recognition of `LIKE` patterns by the planner, the proposed extension allows the type of substring query to be expressed directly in the system catalogue.

Practical value of the obtained results and validation of the method. The proposed method has been validated through a software implementation, SAindex, amounting to 2782 lines of C code and integrated directly into the open-source PostgreSQL 18devel core. The implementation conforms to the standard index-build protocol and performs full write-ahead logging (WAL) of index-page modifications; full support for incremental modifications (`aminsert`, `ambulkdelete`, `amvacuumcleanup`) is identified as a direction for further work. A non-parametric Mann–Whitney U test with effect-size control through Cliff's delta has established the dominance of SAindex over the trigram method `GIN_TRGM` with a median per-pattern speedup of $6.3\times$ for contains queries and $46.5\times$ for prefix queries, an advantage over specialised B-trees under low-selectivity regimes, and on-par performance with `BTREE_REV` under automatic plan selection by the optimiser at the 5M-row scale. The reliability of the experimental results is ensured by a mandatory correctness check (160 of 160 test cases of result-set agreement between SAindex and a sequential scan) and by statistical significance based on 2188 tests of pairwise comparison of methods.

The dissertation consists of five chapters.

In the first chapter, an analytical review of substring-search methods and the theoretical foundations of suffix data structures is conducted. The three target substring-search operations (prefix, suffix, and contains) are formalised; the fundamental suffix structures (suffix tree, suffix array, LCP array, FM-index)

and contemporary algorithms for their construction are reviewed. The available substring-search mechanisms in PostgreSQL and external solutions based on a suffix index are analysed; a fundamental gap is identified between the theoretical capabilities of suffix structures and their realisation in relational DBMSs, which substantiates the research task.

In the second chapter, an experimental evaluation of the existing substring-search mechanisms is carried out: a cross-system comparison of PostgreSQL, Elasticsearch, MySQL, and ClickHouse against a reference suffix-tree implementation, and a detailed benchmark of the PostgreSQL mechanisms identifying the nature of their limitations. A simplified cost model of search in a page-based suffix array in the parameters $\langle B, e, N, L, \mathcal{P} \rangle$ is constructed and the expected advantage is quantitatively estimated.

In the third chapter, a formal model of index access based on a suffix array in external memory in the parameters $\langle B, e, e_k, N \rangle$, independent of any particular DBMS platform, is constructed. The reduction of the three substring-search operations to a single range-search primitive with positional filtering, the bounds of the I/O complexity of the search and collection phases, the completeness of the result-set coverage, and the space complexity of $N(e + 8.125) + B$ bytes are proven.

In the fourth chapter, the design of the index-access method based on a paged suffix array for PostgreSQL is carried out. The PostgreSQL architecture is cast as a system of six invariants C1–C6 that delimit the space of admissible solutions; the six built-in index-access methods are systematically assessed and the unsuitability of each as a carrier of a suffix structure is substantiated. On this basis, a five-section on-disk layout, an index-record format with a materialised key prefix, a five-phase build pipeline driven by `tuplesort`, the `text_sa_ops` operator class with three search strategies, and three TID-collection strategies are proposed. The correctness of the method is proven through a system of five formal statements.

In the fifth chapter, a description of the SAindex software implementation and its experimental evaluation is presented. The implementation is in C, amounting to 2782 lines of code, integrated into the PostgreSQL 18devel core. The methodology of evaluation on three datasets (GeoNames, synthetic_uniform, synthetic_zipf) at five dataset sizes from $100{,}000$ to $5{,}000{,}000$ rows with 30 runs per pattern is formulated. Results of comparing SAindex with `GIN_TRGM`, `BTREE`, and `BTREE_REV` across four pairs are presented; their statistical significance, behaviour under constrained I/O, and the validity of the planner's choice are analysed; the practical limitations of the method

are discussed and directions for further work are outlined.

Keywords: suffix array, trees, B-tree, substring search, natural language processing, index access method, external-memory model, PostgreSQL, pairwise comparisons, Boolean algebra, decision making, evaluation, system architecture, computer system, information technology.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА

Наукові праці, в яких опубліковані основні наукові результати дисертації:

[1] A. Hlybovets and D. Zvazhii, «Implementation of a suffix tree-based index for searching for substrings in a large DBMS,» *Cybernetics and Systems Analysis*, vol. 61, no. 2, pp. 265–275, Mar. 2025, doi: 10.1007/s10559-025-00766-y. *(Scopus, Web of Science Core Collection)*

[2] Д. В. Зважій, «Особливості індексації у PostgreSQL», *Наукові записки НаУКМА. Комп'ютерні науки*, т. 8, с. 113–117, лист. 2025, doi: 10.18523/2617-3808.2025.8.113-117.

[3] Д. В. Зважій та С. С. Гороховський, «Впровадження індексів у POSTGRESQL», *Проблеми програмування*, № 1, с. 14–24, 2026, doi: 10.15407/pp2026.01.014.

[4] Д. В. Зважій та О. М. Малій, «Побудова розподілених суфіксних дерев», *Технічна інженерія*, № 1(97), с. 210–216, лют. 2026, doi: 10.26642/ten-2026-1(97)-210-216.

Наукові праці, які засвідчують апробацію матеріалів дисертації:

[1] Д. В. Зважій, «Використання суфіксних дерев для покращення операцій пошуку в системах управління базами даних», у *Матеріали XII Всеукраїнської наукової конференції молодих математиків*, Київ, 9–11 трав. 2024. URL: https://ekmair.ukma.edu.ua/items/fe4ef6c9-d05c-459c-b26e-d68e6e288c8e

[2] Д. В. Зважій, «Перспективи впровадження індексу на базі суфіксного дерева для пошуку підрядків у СУБД великого розміру», у *Матеріали XVI Міжнародної науково-практичної конференції «Теоретичні та прикладні аспекти побудови програмних систем» TAAPSD'2025*, Київ, 24–25 лист. 2025. URL: https://ekmair.ukma.edu.ua/items/b25fd968-b4af-42f2-ab8d-b3f5f45cdc13

[3] Д. В. Зважій, «Застосування суфіксних масивів для розробки індексів у PostgreSQL», у *Матеріали Міжнародної науково-практичної конференції здобувачів вищої освіти і молодих вчених «Інформаційні технології: теорія і практика»*, Харків, 25–27 бер. 2026. URL: https://knit.kname.edu.ua/home/science/internet-conference