



ДОРОШЕНКО

Анатолій Юхимович —
доктор фізико-математичних
наук, завідувач відділу теорії
комп'ютерних обчислень
Інституту програмних систем
НАН України

ФОРМАЛЬНІ ТА ІНТЕЛЕКТУАЛЬНІ МЕТОДИ АВТОМАТИЗАЦІЇ ПРОЄКТУВАННЯ ПРОГРАМНИХ СИСТЕМ

**Стенограма доповіді на засіданні
Президії НАН України 19 лютого 2025 року**

У доповіді обговорено результати робіт, виконаних в Інституті програмних систем НАН України, з розвитку формальних та інтелектуальних методів автоматизації проектування програмних систем, які сьогодні широко використовують у галузі безпеки та оборони, метеорології, екології, енергетиці, нанотехнологіях, робототехніці тощо. Наголошено, що українська наукова школа з цього напрямку, започаткована ще академіком В.М. Глушковим, сьогодні займає передові позиції у світі. Розвиток формальних та інтелектуальних методів автоматизації проектування програмних систем досяг необхідної наукової і технологічної зрілості, щоб стати основою цифрової трансформації галузей економіки України.

Шановний пане президенте!

Шановні члени Президії!

Розпочну свою доповідь з розгляду проблеми розроблення програмного забезпечення. Тривалий час вважали, що програмний продукт є таким самим виробом, як і продукти, що виробляються в інших галузях економіки. Ще й зараз серед нефахівців побутує уявлення, що розроблення програмного забезпечення — це просто програмування, тобто виробництво. Однак сучасна парадигма програмної інженерії як науки стверджує, що програмний продукт не виробляють, а *розробляють*. На перший погляд, це ніби незначна лінгвістична відмінність, але вона має глибокий семантичний сенс.

Проблема розроблення програмного забезпечення полягає в дотриманні умов тріади для будь-якого проекту: 1) забезпечення необхідної якості програмного забезпечення; 2) реалізація за обмежений проміжок часу; 3) виконання проекту в межах визначеного бюджету. Статистичні дані в галузі програмної інженерії свідчать про те, що у світі значна частина програмних проєктів (приблизно три з чотирьох) завершується невдачею.

Це означає, що вони або не вкладаються в бюджет, або виходять за часові рамки, або не відповідають початково поставленим вимогам. Більше того, залежність вартості й тривалості проекту від його складності (в людино-місяцях) має нелінійний, а часто навіть експоненційний характер.

Особливо важливою проблема розроблення програмного забезпечення стає у сфері сучасних складних обчислювальних систем паралельної дії, зокрема суперкомп'ютерного класу. Традиційні компіляторні технології для таких систем мають обмежене значення, а процеси розроблення містять велику частку ручної праці. До того ж генерований код часто є не оптимальним, що призводить до недовикористання потенційної потужності мультипроцесорних систем.

Отже, основний виклик стосується проблеми *автоматизації* генерації ефективного коду для паралельних обчислювальних систем складної архітектури, що охоплюють велике розмаїття мультипроцесорних платформ. Фундаментальним підходом для розв'язання цієї проблеми є автоматизація процесу проектування програмного забезпечення на основі *формалізації* предметних галузей та *інтелектуалізації* засобів трансформації початкових специфікацій у вихідний код певною мовою програмування.

Наукові основи формальних методів автоматизації проектування програмних систем було закладено ще академіком Віктором Михайловичем Глушковым, який запропонував ідею використання алгебри алгоритмів як засобу для специфікації, трансформації та оптимізації комп'ютерних програм. Його піонерська стаття, опублікована в 1965 р., стала першою у світі роботою з цього напрямку.

У наступні роки в Інституті кібернетики імені В.М. Глушкова НАН України, Київському національному університеті імені Тараса Шевченка та інших провідних наукових центрах було виконано цілу низку масштабних проєктів. У 1970-х — 1980-х роках проведено комплекс науково-дослідних робіт з розроблення теорії та створення кількох алгебро-алгоритмічних

інструментів автоматизації проектування апаратного та програмного забезпечення обчислювальних систем, зокрема розроблено такі системи, як «Проект», «Мультипроцесист», «Парус», макроконвеєрний обчислювальний комплекс APS та ін.

Сформувалася наукова школа формальних методів проектування програмного забезпечення, до складу якої увійшли академіки НАН України О.А. Летичевський, П.І. Андон, А.В. Анісімов; члени-кореспонденти НАН України К.Л. Ющенко, О.Л. Перевозчикова; доктори наук Ю.В. Капітонова, Г.О. Цейтлін, І.В. Вельбицький, А.Ю. Дорошенко, М.С. Нікітченко, С.Л. Кривий, С.Д. Погорілий, О.О. Летичевський та ін.

В Інституті програмних систем НАН України цей напрям досліджень від самого початку був спрямований на розвиток високопродуктивних обчислень на різних платформах паралельної обробки — від багатоядерних процесорів до мультипроцесорних кластерів та хмарних середовищ.

Сама ідея алгебри алгоритмів полягала в її двоосновності, тобто поєднанні двох алгебр: алгебри умов і алгебри операторів через операцію множення умови на оператор і далі у створенні універсального набору конструкцій (з операторів композиції, альтернативи і циклу), за допомогою якого можна описати будь-який алгоритм, що існує в природі.

Для реалізації цієї ідеї було розроблено методологію і алгебро-алгоритмічний інструментарій для автоматизації проектування програм, які на основі алгебри алгоритмів охоплюють теорію і методологію алгебри алгоритмів у предметних галузях та формальні специфікації алгебро-алгоритмічних конструкцій для конкретних класів алгоритмів. Разом вони забезпечують практичну базу для автоматизованого проектування (паралельних) програм на основі програмних трансформацій від високорівневих специфікацій аж до нижнього рівня мови програмування. З використанням цього підходу розробляють програмне забезпечення для паралельних платформ, таких як мультиядерні центральні і графічні процесори, кластери, грі-

Рис. 1. Алгебро-алгоритмічний інструментарій для автоматизації проектування паралельних програм



ди, «хмарні» архітектури тощо, на яких можна створювати широкі класи предметно-орієнтованих застосунків (рис. 1).

Отже, методи алгебри алгоритмів та інструменти їх автоматизованого застосування дозволили створити потужну платформу для автоматизації розроблення програмних систем.

У чому ж полягають переваги формальних методів автоматизації паралельного програмування і проектування?

По-перше, формальні методи розроблення не залежать від цільової мови програмування та частково не залежать і від мультипроцесорних платформ, тобто вони мають високий рівень і можуть бути специфіковані для будь-яких кінцевих інструментальних мов програмування.

По-друге, перевага використання програмних моделей високого рівня взагалі полягає в більш короткому та виразному записі початкових специфікацій.

По-третє, програми, виконані з використанням цього інструментарію, гарантують принаймні синтаксичну, а в багатьох випадках і семантичну коректність розроблюваних програм.

По-четверте, автоматизовані трансформації програмних артефактів у процесі розроблення значно скорочують обсяги ручної праці і

можуть забезпечити потужні перетворення завдяки багаторазовому застосуванню одного набору правил до багатьох частин коду, хоча самі перетворення все ж проектує людина, тобто це все ще людиноцентрична технологія.

У 1990-х роках світовою тенденцією для досягнення вищого ступеня автоматизації проектування програмних систем стало створення механізмів автоматичного виконання програмних перетворень на основі застосування шаблонів трансформацій, або так званих правил переписування термів. Так з'явилася нова *алгебраїчна парадигма* програмування, така ж універсальна, як і імперативна, функціональна чи об'єктно-орієнтована, але в основу якої покладено механізм програмного рушія з автоматичним виконанням переписувальних правил.

Українська наукова школа за цим напрямом також була однією з перших у світі, а в Інституті програмних систем НАН України через кілька років під керівництвом доктора фізико-математичних наук А.Ю. Дорошенка було розроблено систему програмування переписувальних правил під назвою TermWare. Її особливістю є можливість вбудовування в уже наявні прикладні програмні системи.

Важливим об'єктом застосування TermWare стала система автоматизації проектування ав-

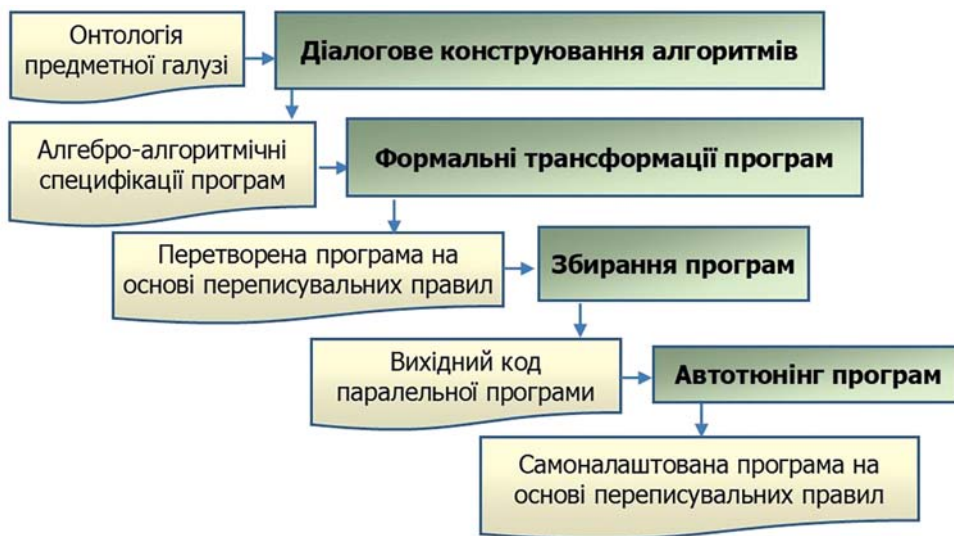


Рис. 2. Схема технологічної лінії застосування формальних і адаптивних методів автоматизації проектування програм

тотюнерів — засобів самоналаштування програмного забезпечення для високопродуктивних обчислень [1].

Однією з яскравих ілюстрацій практичної вигоди від застосування TermWare є використання формальних трансформацій у задачах чисельного прогнозування погоди. Детальніше про це я розповім далі, а тут лише зазначу, що коли ми застосували автоматизовану трансформацію тривимірних масивів в одновимірні, це забезпечило мультипроцесорне прискорення більш як на порядок(!).

Загалом у нашому інституті прийшли до такої схеми технологічної лінії застосування формальних і адаптивних методів автоматизації проектування програм (рис. 2). Розроблення починається з предметної галузі, точніше з її онтології, де ми аналізуємо концепти цієї предметної галузі, їхні взаємозв'язки тощо. Далі за допомогою алгебро-алгоритмічної специфікації створюємо специфікацію найважливіших алгоритмів, які працюють у цій предметній галузі. Потім через формальні трансформації за допомогою переписувальних правил ми перетворюємо ці програми і вибираємо вихідний код паралельної програми вже мовою програмування. На завершальному етапі виконуємо самоналаштування на основі знову ж таки переписувальних правил, які оптимізують вихідний код.

Однак розглянуті вище методи автоматизації, навіть у їх адаптивному варіанті, є людиноцентричними, тобто все залежить від розробника, його кваліфікації, досвіду, навичок. Тому ці методи ще не можна назвати інтелектуальними, адже вони працюють з повністю визначеною інформацією щодо прийняття рішень у процесі проектування.

Інтелектуальні методи, зокрема штучний інтелект (ШІ), вступають у дію тоді, коли розробнику доводиться спиратися на *неоднозначну, неповну або нечітку* інформацію, і для прийняття рішень потрібно проявляти риси, притаманні людському інтелекту, такі як досвід, інтуїція, передбачення тощо.

Типовими засобами ШІ є штучні нейронні мережі (ШНМ) та методи машинного навчання для розв'язання складних задач управління. Особливо цікавим у цьому плані є підхід нейроеволюції, який дозволяє здійснити глибоку автоматизацію розроблення ШНМ за допомогою генетичних алгоритмів. При цьому одночасно розробляються і топологія, і вагові коефіцієнти зв'язків для штучних нейромереж.

В Інституті програмних систем НАН України ми вдосконалили метод нейроеволюції завдяки розпаралелюванню генетичних алгоритмів та поширенню алгебро-алгоритмічного інструментарію на предметну галузь машинного навчання ШНМ.

Отже, *нейроеволюція топологій, що наростають* (NEAT), — це генетичний алгоритм знаходження за допомогою еволюційного пошуку ШНМ й утвердження найбільш вдалих варіантів видоутворень нейромереж. При цьому одночасно змінюються і топологія, і значення ваги з'єднань між вузлами штучної нейронної мережі. Інакше кажучи, NEAT — це генетичний алгоритм, який шляхом *схрещування та мутацій* автоматично продукує інші топології і оцінює їх за допомогою *функції придатності* (fitness) з метою відбору найкращих для продовження генерації поколінь.

Метод нейроеволюції топологій, що наростають, особливо добре відповідає потребам *машинного навчання з підкріпленням*. А це найскладніший клас машинного навчання, в якому немає вчителя.

Розроблені в Інституті програмних систем НАН України формальні та інтелектуальні методи автоматизації проектування програмних систем уже впроваджено в практичну діяльність у галузі метеорології, екології, енергетики, у сфері безпеки і оборони, в нанотехнологіях, робототехніці тощо. Нашими партнерами й користувачами є, зокрема, Державна служба спеціального зв'язку та захисту інформації України, Український гідрометеорологічний інститут ДСНС України та НАН України, Державне спеціалізоване підприємство «Екоцентр», Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, Тернопільський національний технічний університет імені Івана Пулюя, АТ «Антонов», Міністерство оборони України та інші організації й відомства.

Так, наш інститут брав участь у розробленні захищеної платформи Defense Resource Micro Services (DRMS), створеної на замовлення Державної служби спеціального зв'язку та захисту інформації України. Це платформа мікросервісів, яка орієнтована на типові потреби інформаційно-аналітичних систем оборонного призначення з розширеними можливостями комплексного захисту інформації. Завдяки цьому платформу можуть використовувати не лише військові, а й державні установи.

Ми маємо давні й тісні робочі зв'язки з Українським гідрометеорологічним інститутом ДСНС України і НАН України. Свого часу ми спільно розробили спосіб паралельної реалізації чисельного регіонального прогнозування погоди. Проект тоді було підтримано грантом наукової програми НАТО. А тепер маємо ще одну спільну розробку. В результаті проведених досліджень, метою яких була автоматизація прогнозування погоди та якості повітря засобами машинного навчання, створено загальнодоступний вебсервіс, який на запит прогнозних даних здійснює збирання ретроспективних даних про погоду та забруднення повітря; в реальному часі тренує алгоритм машинного навчання й оптимізує його гіперпараметри; виводить на мобільні пристрої результати прогнозування у вигляді інтерактивного графіка. При цьому точність прогнозування сягає 97—99 % [2]. Це дуже високий показник, оскільки традиційні чисельні методи прогнозування погоди мають набагато нижчу точність.

Ще один приклад — прогнозування споживання електроенергії в Україні на основі машинного навчання. Добре відомо, наскільки важливою є проблема безпеки в енергетичному секторі, а під час війни це стало фактором виживання України. Метою управління та моніторингу в цій сфері є оптимізація перетоків електроенергії між учасниками ринку, тобто між європейськими партнерами та Україною, для підтримання балансу між виробниками і споживачами енергії.

За сприяння фахівців Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України ми розробили програмну систему прогнозування споживання електроенергії з використанням методів машинного навчання. Результати машинного навчання із застосуванням методів регресії показали, що можна з великою точністю (до 95 %) спрогнозувати тарифні плани на одному з основних сегментів енергоринку — ринку на добу наперед, та вийти за межі одноденного планування [3].

Останній приклад застосувань, який я встигаю навести, це прогнозування радіонуклідно-

го забруднення вод Дніпровського каскаду засобами машинного навчання.

Радіоактивне забруднення найбільшої водної артерії України р. Дніпро є загрозою для населення, що проживає в її басейні. У нашому інституті розроблено нову камерну модель агрегованої мертвої зони, яка дає змогу адекватно відтворювати концентрації розчинених речовин із високим ступенем точності. Крім того, розроблено метод прогнозування концентрації радіонуклідів засобами машинного навчання. За сприяння Державного спеціалізованого підприємства «Екоцентр» виконано верифікацію методу на ретроспективних даних забруднення вод у післячорнобильські роки. Разом із партнерами ми подали проектну заявку до грантової програми «Горизонт Європа» [4].

Останніми роками за тематикою розвитку наших розробок із застосування формальних методів та інтелектуалізації засобів розроблення програмного забезпечення для прикладних галузей ми видали 8 монографій, 4 з яких ан-

глійською мовою, та кілька підручників для вищої школи [5—12]. Маємо також охоронні документи у вигляді авторських свідоцтв на розроблені нами програмні комплекси.

Отже, українська наукова школа з формальних методів розроблення/алгебраїчного програмування, започаткована свого часу академіком В.М. Глушковим, і сьогодні посідає передові позиції у світі, збагачуючись інтелектуалізацією своїх інструментів та захоплюючи «нові території» у прикладних галузях застосувань.

Формальні та інтелектуальні методи автоматизації проектування взаємно доповнюють одні одних і йдуть пліч-о-пліч з використанням великих даних та їх обробленням за допомогою паралельних обчислень.

Ці напрями сьогодні вже досягли необхідної наукової і технологічної зрілості, щоб становити основу цифрової трансформації галузей економіки України.

Дякую за увагу!

За матеріалами засідання підготувала О.О. Мележик

REFERENCES

[СПИСОК ЛІТЕРАТУРИ]

- Ivanenko P.A., Doroshenko A.Yu. Method of Automated Generation of Autotuners for Parallel Programs. *Cybernetics and Systems Analysis*. 2014, **50**(3): 465—475. <https://doi.org/10.1007/s10559-014-9635-3>
- Doroshenko A.Yu., Haidukevych Ya.O., Haidukevych V.O., Zhyrenkov O.S. User-centric technology stack for weather and air pollution forecasting. *Problems in Programming*. 2024. (4): 34—42. <https://doi.org/10.15407/pp2024.04.034> [Дорошенко А.Ю., Гайдукевич Я.О., Гайдукевич В.О., Жиренков О.С. Клієнто-центричний технологічний стек для прогнозу погоди та якості повітря. *Проблеми програмування*. 2024. № 4. С. 34—42.]
- Doroshenko A., Zhora D., Haidukevych V., Haidukevych Y., Yatsenko O. Predicting 24-Hour Nationwide Electrical Energy Consumption Based on Regression Techniques. *CEUR-WS*. 2024. **3806**: 122—138.
- Doroshenko A., Sizonenko V., Zhora D., Yatsenko O. Forecasting the Dnipro Water Reservoir Cascade Pollution using Machine Learning Methods. In: *Mathematical Modeling and Simulation of Systems 2024*. Proc. 19th Int. Conf. Chernihiv, Ukraine (in print).
- Doroshenko A.E., Zhareb K.A., Ivanov E.V., Nikitchenko N.S., Yatsenko E.A. *Formalnyye metody postroyeniya parallelnykh programm* [Formal methods for constructing parallel programs]. Кропивницький, 2016 [in Russian]. [Дорошенко А.Е., Жереб К.А., Иванов Е.В., Никитченко Н.С., Яценко Е.А. *Формальные методы построения параллельных программ*. Кропивницький, 2016.]
- Andon F.I., Doroshenko A.E., Zhareb K.A., Shevchenko R.S., Yatsenko E.A. *Metody algebraicheskogo programmirovaniya; formalnyye metody razrabotki parallelnykh programm* [Methods of algebraic programming; formal methods for developing parallel programs]. Kyiv: Naukova Dumka, 2017 [in Russian]. [Андон Ф.И., Дорошенко А.Е., Жереб К.А., Шевченко Р.С., Яценко Е.А. *Методы алгебраического программирования; формальные методы разработки параллельных программ*. Киев: Наукова думка, 2017.]
- Prusov V., Doroshenko A. *Computational Techniques for Modeling Atmospheric Processes*. Hershey: IGI Global, 2018. <https://doi.org/10.4018/978-1-5225-2636-0>

8. Doroshenko A.Yu., Andon P.I., Yatsenko O.A., Zhreb K.A. *Algebra-Algorithmic Models and Methods of Parallel Programming*. Kyiv: Akadempriodyka, 2018. <https://doi.org/10.15407/akadempriodyka.367.192>
9. Doroshenko A.Yu., Ivanov Ie.V., Nikitchenko M.S., Yatsenko O.A., Zhreb K.A. *Formal Program Development Methods*. Kyiv: Publishing and Printing Center “Kyiv University”, 2021.
10. Shpyg V., Doroshenko A., Shcheglov O. et al. *Eksabaytova ekonomika [Exabyte Economy]*. Edited by I.L. Tatomyr, Yu.O. Shulzhik. Truskavets: Posvit, 2022 [in Ukrainian].
[Шпиг В., Дорошенко А., Щеглов О. та ін. *Ексабайтова економіка*. За наук. ред. І.Л. Татомир, Ю.О. Шульжика. Трускавець: Посвіт, 2022.]
11. Doroshenko A., Yatsenko O. *Formal and adaptive methods for automation of parallel programs construction: emerging research and opportunities*. Hershey: IGI Global, 2021. <https://doi.org/10.4018/978-1-5225-9384-3>
12. Andon P.I., Doroshenko A.Yu., Akulovsky V.G., Ivanenko P.A., Yatsenko O.A. *Formalni ta adaptivni metody konstruivannya vysokoproduktyvnykh paralelnykh prohram [Formal and adaptive methods for constructing high-performance parallel programs]*. Kyiv: Naukova Dumka, 2023 [in Ukrainian]. <https://doi.org/10.15407/978-966-00-1809-9>
[Андон П.І., Дорошенко А.Ю., Акуловський В.Г., Іваненко П.А., Яценко О.А. *Формальні та адаптивні методи конструювання високопродуктивних паралельних програм*. Київ: Наукова думка, 2023.]

Anatoliy Y. Doroshenko

Institute of Software Systems of the National Academy of Sciences of Ukraine, Kyiv, Ukraine

ORCID: <https://orcid.org/0000-0002-8435-1451>

FORMAL AND INTELLIGENT METHODS FOR AUTOMATING THE DESIGN OF SOFTWARE SYSTEMS

Transcript of scientific report at the meeting of the Presidium of NAS of Ukraine, February 19, 2025

The report discusses the results of the works carried out at the Institute of Software Systems of the National Academy of Sciences of Ukraine on the development of formal and intelligent methods for automating the design of software systems, which are widely used in security and defense, meteorology, ecology, energy, nanotechnology, robotics, etc. It is emphasized that the Ukrainian scientific school in this area, founded by Academician V.M. Glushkov, is now at the forefront in the world. The development of formal and intelligent methods for automating the design of software systems has reached the necessary scientific and technological maturity to become the basis for the digital transformation of Ukraine's economic sectors.

Cite this article: Doroshenko A.Y. Formal and intelligent methods for automating the design of software systems (transcript of scientific report at the meeting of the Presidium of NAS of Ukraine, February 19, 2025). *Visn. Nac. Akad. Nauk Ukr.* 2025. (4): 63—69. <https://doi.org/10.15407/visn2025.04.063>