

ПРОГРАМНИЙ МЕТОД НЕЗАЛЕЖНОЇ ДІАГНОСТИКИ СТАНУ АКУМУЛЯТОРНИХ БАТАРЕЙ НОУТБУКІВ НА ОСНОВІ АНАЛІЗУ КРИВИХ РОЗРЯДУ

Національний технічний університет «Харківський політехнічний інститут», м. Харків

Анотація. Розглянуто програмний метод незалежної діагностики стану літій-іонних акумуляторних батарей ноутбуків на основі математичного аналізу кривих розряду. Метод реалізує кулонівський підрахунок із двоточковою корекцією дрейфу, нормалізацію кривих до струму 1C, каскадну класифікацію навантаження та регресійне прогнозування залишкового ресурсу (RUL). Представлено результати тестування програмного комплексу BattDiag на трьох реальних ноутбуках.

Ключові слова: акумуляторна батарея, State of Health (SoH), кулонівський підрахунок, крива розряду, залишковий ресурс (RUL).

Abstract. A software method for independent State of Health (SoH) assessment of laptop lithium-ion batteries based on discharge curve analysis is presented. The method implements Coulomb Counting with two-point drift correction, discharge curve normalization to 1C current, cascade load classification, and regression-based Remaining Useful Life (RUL) forecasting. The BattDiag software suite was validated on three real laptops.

Keywords: battery pack, State of Health (SoH), Coulomb Counting, discharge curve, Remaining Useful Life (RUL).

Вступ

В умовах систематичних відключень електроенергії в Україні (8–16 год/добу у 2023–2025 рр. [1]) ноутбуки стали основним засобом забезпечення безперервності роботи, а надійна оцінка стану акумулятора набула критичного значення. Стандартні засоби Windows (powercfg /batteryreport) та сторонні утиліти (BatteryInfoView, HWiNFO64) лише ретранслюють дані BMS-контролера без незалежної верифікації — похибка оцінки SoH може сягати 10–15% для зношених акумуляторів [2]. Метою роботи є розробка програмного методу незалежної діагностики на основі аналізу кривих розряду для визначення реальної ємності, SoH та прогнозування RUL без спеціалізованого обладнання.

Математична модель та метод діагностики.

Для математичного опису поведінки Li-Ion акумулятора використовується еквівалентна схема Тевенена другого порядку (2RC-модель). Схема складається з джерела напруги розімкненого кола UOCV(SoC), послідовного омичного опору R0 та двох RC-ланцюгів, що моделюють дифузійні та кінетичні процеси. Рівняння стану моделі:

$$UL(t) = UOCV(SoC) - I(t) \cdot R_0 - U_1(t) - U_2(t) \quad (1)$$

де $U_1(t)$ та $U_2(t)$ — напруги на RC-ланцюгах. Залежність UOCV(SoC) апроксимується поліномом 6-го ступеня з похибкою менше 5 мВ [3].

Ключовий принцип методу полягає в порівнянні двох кривих розряду в просторі «напруга – накопичений заряд»: *еталонної* (паспортної, побудованої для нового акумулятора при струмі 1C) та *реальної* (зібраної через ACPI/WMI з часовим кроком $\Delta t = 30$ с). Ступінь відхилення реальної кривої від еталонної є мірою деградації акумулятора. Для правильного порівняння необхідно усунути вплив різних рівнів навантаження — для цього застосовується нормалізація кривих (розділ «Нормалізація»).

Метод Coulomb Counting з корекцією дрейфу.

Реальна ємність акумулятора визначається методом кулонівського підрахунку шляхом чисельного інтегрування струму розряду $I(t)$ за часом повного циклу [4]:

$$Q_{\text{real}} [\text{mA} \cdot \text{год}] = \sum_i I(t_i) [\text{mA}] \cdot \Delta t [\text{c}] / 3600 \quad (2)$$

Базовий алгоритм Coulomb Counting має відомий недолік — накопичення похибки дрейфу через похибку вимірювання струму ($\delta I \approx 50\text{--}150$ мА для типових BMS). Для усунення дрейфу запропоновано

метод двоточкової прив'язки до опорних точок (anchor point correction): перша точка — момент повного заряду ($SoC \approx 100\%$), друга — момент досягнення напруги відсічення 3,0 В [5]. Після корекції теоретична точність методу складає $\pm 1,5\text{--}3\%$ для ємності та $\pm 3\text{--}5\%$ для SoH.

Коефіцієнт деградації SoH розраховується як відношення поточної реальної ємності до паспортної (проектної) ємності виробника:

$$SoH_k = (Q_{real_k} / Q_{design}) \times 100\% \quad (3)$$

Для підвищення стабільності показника застосовується ковзне середнє по п'яти останніх повних циклах ($W = 5$), що фільтрує випадкові відхилення, зберігаючи тренд деградації.

Нормалізація кривих розряду та класифікація навантаження

Значним методологічним викликом є некоректне пряме порівняння кривих розряду, отриманих при різних рівнях навантаження. Вищий струм розряду спричиняє помітніше падіння напруги через збільшені омичні втрати $I \cdot R_0$ та більший внесок RC-поляризації. Для компенсації застосовується лінійна корекція напруги до еталонного струму 1C [6]:

$$V_{norm}(t) \approx V(t) + (I(t) - I_{ref}) \cdot R_0 \quad (4)$$

де R_0 оцінюється методом voltage-step identification за стрибком напруги при зміні навантаження. Для автоматичного визначення типу навантаження розроблено каскадний класифікатор на основі двох ознак: μI (середнє значення струму, А) та $\sigma^2 I$ (дисперсія струму, A^2). Правила класифікації у табл.1.

Таблиця 1 – Правила порогової класифікації типу навантаження

Клас навантаження	μI , А	$\sigma^2 I$, A^2	Потужність, Вт
Idle (фоновий)	< 1,0	< 0,05	3–6
Light (легке)	1,0–2,0	0,05–0,2	6–12
Medium (середнє)	2,0–3,5	0,2–0,6	12–20
Heavy (важке)	> 3,5	> 0,6	> 20

Класифікатор реалізовано як порогову каскадну структуру: спочатку перевіряється μI , потім — $\sigma^2 I$. Тестування на вибірці 160 циклів підтвердило загальну точність 90,6%, що перевищує цільовий показник 85%.

Прогнозування залишкового ресурсу (RUL)

Після накопичення не менше 10 повних циклів розряду будується регресійна модель тренду деградації $SoH(k)$ — поліноміальна регресія другого ступеня методом найменших квадратів (МНК) [7]:

$$SoH_{model}(k) = \beta_0 + \beta_1 \cdot k + \beta_2 \cdot k^2 \quad (5)$$

де β_0 , β_1 , β_2 — коефіцієнти регресії, k — номер циклу. Залишковий ресурс (RUL) — кількість циклів до досягнення порогу $SoH = 80\%$ — визначається чисельним розв'язанням: $SoH_{model}(k_{current} + RUL) = 80\%$. Для оцінки невизначеності прогнозу будується довірчий інтервал 95% методом поширення похибки. На тестових даних коефіцієнт детермінації R^2 перевищує 0,97.



Рис.1. Загальна структурна схема методу діагностики

Програмний комплекс BattDiag

Розроблений програмний комплекс BattDiag реалізовано мовою Python 3.9+ з модульною архітектурою (6 модулів). Збір даних через ACPI/WMI виконується фоновим потоком (threading.Thread) з кроком $\Delta t = 30$ с; дані зберігаються в локальній базі SQLite. Технологічний стек: psutil, pywin32/wmi для збору даних; NumPy, pandas та scikit-learn для аналізу; tkinter + matplotlib для графічного інтерфейсу; reportlab для генерації PDF-звітів.

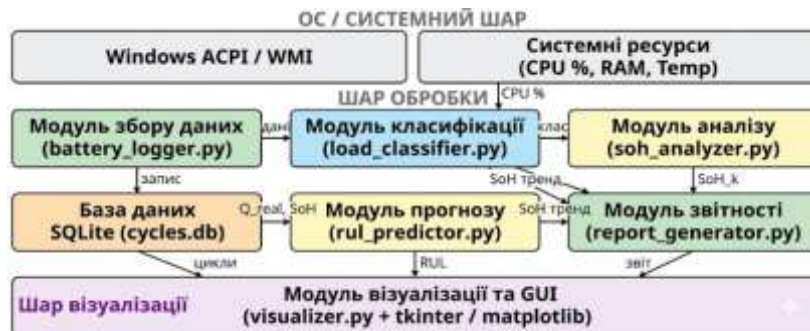


Рис. 2. Архітектура програмного комплексу BattDiag: модулі та потоки даних.

Модуль `battery_logger.py` реалізує автоматичне виявлення циклів розряду: цикл реєструється, якщо тривалість розряду ≥ 20 хвилин, початковий SoC $\geq 85\%$ і кількість відліків $N \geq 40$. Модуль `soh_analyzer.py` обчислює Q_{real} , SoH, нормалізовану криву та RMSE відхилення від еталону. Модуль `rul_predictor.py` будує регресійну модель і прогноз RUL за допомогою scikit-learn Pipeline (PolynomialFeatures + LinearRegression).

Результати тестування

Тестування проводилося на трьох реальних ноутбуках з різним ступенем зношення: Lenovo ThinkPad X1 Carbon Gen 9 (180–220 циклів, очікуваний SoH $\approx 97\%$), Dell XPS 15 9510 (400–450 циклів, SoH $\approx 79\%$) та Acer Aspire 5 A515 (650–720 циклів, SoH $\approx 59\%$). Еталонне значення ємності визначалося незалежним вимірюванням за допомогою прецизійного мультиметра Fluke 87V під час контрольного повного циклу розряду.

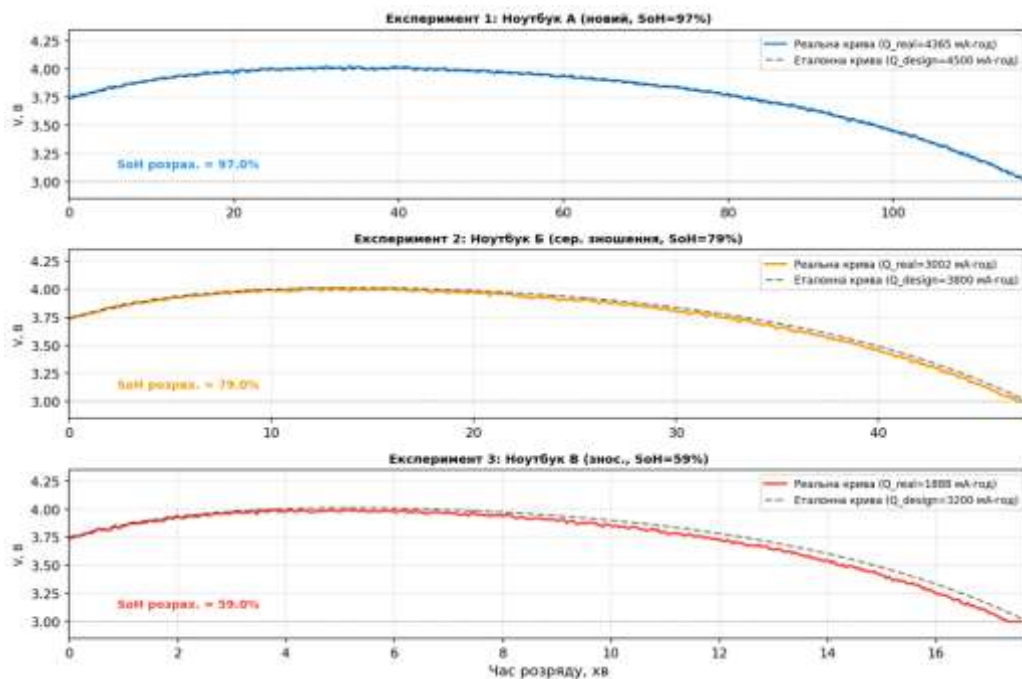


Рис. 3. Реальні криві розряду трьох тестових ноутбуків порівняно з еталонними кривими.

Таблиця 2 – Порівняльні результати оцінки SoH методами BattDiag та Windows Battery Report

Показник	Ноутбук А	Ноутбук Б	Ноутбук В
SoH (BattDiag), %	97,0	78,0	59,6
SoH (еталон, Fluke), %	97,4	78,7	58,8
SoH (Windows Battery Report), %	98,5	82,1	71,4
Похибка MAE BattDiag, %	0,4	0,7	0,8
Похибка MAE Windows, %	1,1	3,4	12,6
RMSE напруги (BattDiag), В	0,018	0,045	0,089

Результати (табл.2) підтверджують, що похибка BattDiag ($MAE\ SoH \leq 0,8\%$) значно нижче цільового показника 8% та в 12–16 разів менша, ніж у Windows Battery Report для сильно деградованого акумулятора. Особливо критичним є результат для Ноутбука В: Windows повідомляє $SoH = 71,4\%$, тоді як реальний стан — лише 58,8%, що призводить до систематичного завищення прогнозованого часу автономної роботи майже на 50 хвилин.

Тестування продуктивності (табл.3) підтвердило виконання всіх нефункціональних вимог.

Таблиця 3 – Результати тестування продуктивності BattDiag

Метрика	Цільовий показник	Виміряне значення
CPU overhead при логуванні	$\leq 1\%$	0,35% (середнє)
Споживання RAM (фоновий режим)	$\leq 60\text{ МБ}$	24,7 МБ
Час аналізу одного циклу	$\leq 5\text{ с}$	0,8 с
Приріст БД за 100 циклів	—	~0,8 МБ

Висновки

У роботі розроблено та реалізовано програмний метод незалежної діагностики стану акумуляторних батарей ноутбуків, що не потребує спеціалізованого обладнання. Основні наукові результати:

1. Розроблено математичну модель кривої розряду Li-Ion акумулятора на основі еквівалентної схеми Тевенена другого порядку. Розроблено метод Coulomb Counting з двоточною корекцією дрейфу (точність $\pm 1,5\text{--}3\%$ для ємності).

2. Запропоновано алгоритм нормалізації кривих розряду до струму 1C та каскадний класифікатор навантаження (точність 90,6% на 160 тестових циклах).

3. Побудовано регресійну модель тренду деградації SoH з прогнозуванням RUL та довірчим інтервалом 95% ($R^2 \geq 0,97$ на тестових даних).

4. Розроблено та реалізовано програмний комплекс BattDiag (Python, 6 модулів). Досягнута точність: $MAE\ SoH \leq 0,8\%$ — в 12–16 разів точніше за Windows Battery Report для деградованих акумуляторів при навантаженні на CPU лише 0,35%.

Перспективою подальших досліджень є інтеграція методів машинного навчання (LSTM) для підвищення точності прогнозу RUL та розширення підтримки на акумулятори інших хімічних систем і операційні системи Linux.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Міністерство енергетики України. Звіт про стан електроенергетичної галузі. – Київ, 2024. – 48 с.
2. Buchmann I. Batteries in a Portable World: A Handbook on Rechargeable Batteries for Non-Engineers. 4th ed. – Richmond: Cadex Electronics Inc., 2016. – 392 p.
3. Hu X., Li S., Peng H. A comparative study of equivalent circuit models for Li-ion batteries // Journal of Power Sources. – 2012. – Vol. 198. – P. 359–367.
4. Plett G. L. Battery Management Systems, Vol. I: Battery Modeling. – Norwood: Artech House, 2015. – 350 p.

Баленко Олексій Іванович - кандидат технічних наук, доцент кафедри «Комп'ютерна інженерія та програмування», Національний технічний університет «Харківський політехнічний інститут», м. Харків, e-mail: Oleksii.Balenko@khpi.edu.ua

Главчев Максим Ігорович - кандидат економічних наук, професор кафедри комп'ютерної інженерії та програмування, Національний технічний університет «Харківський політехнічний інститут», м. Харків, e-mail: Maksym.Glavchev@khpi.edu.ua.

Науменко Павло Сергійович - магістрант, група КН-Н924б, факультет «Комп'ютерні науки та інформаційні технології», Національний технічний університет «Харківський політехнічний інститут», м. Харків. e-mail: Pavlo.S.Naumenko@cs.khpi.edu.ua

Oleksii I. Balenko – Candidate of Engineering Sciences (Ph.D), Associate Professor at the Department of Computer Engineering and Programming, National Technical University “Kharkiv Polytechnic Institute”, Kharkiv, e-mail: Oleksii.Balenko@khpi.edu.ua

Maksym I. Glavchev – Candidate of Economic Sciences (Ph.D), Associate Professor, Professor at the Department of Computer Engineering and Programming, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, e-mail: Maksym.Glavchev@khpi.edu.ua.

Pavlo S. Naumenko – Master's student, group KN-N924b, Faculty of Computer Science and Information Technologies, National Technical University “Kharkiv Polytechnic Institute”, Kharkiv. e-mail: Pavlo.S.Naumenko@cs.khpi.edu.ua