Стахів П. Розпаралелення процесу побудови дискретної динамічної моделі двообмоткового трансформатора / Стахів П., Струбицька І., Козак Ю. // Вісник ТНТУ. — 2012. — Том 65. — № 1. — С.182-187. — (математичне моделювання. математика. фізика).

УДК 519.688

П. Стахів[1], докт. техн. наук; І. Струбицька[2];
Ю. Козак[1], канд. техн. наук

[1]Національний університет «Львівська політехніка»
[2]Тернопільський національний економічний університет

РОЗПАРАЛЕЛЕННЯ ПРОЦЕСУ ПОБУДОВИ ДИСКРЕТНОЇ ДИНАМІЧНОЇ МОДЕЛІ ДВООБМОТКОВОГО ТРАНСФОРМАТОРА

Резюме. *Розпаралелено процес побудови дискретної динамічної моделі двообмоткового трансформатора. Для розпаралелення вибрано обчислювальну архітектуру SIMD, а засобом розпаралелення – графічний багатоядерний процесор. За допомогою програми для експериментального набору побудовано лінійну модель двообмоткового трансформатора. Для перевірки точності створеної моделі було проведено порівняння значень, отриманих за допомогою моделі, з фактичними значеннями. Проведено чисельні експерименти, які підтвердили ефективність розпаралелення даної задачі, а саме – порівняно час виконання задачі на центральному і графічному процесорних елементах.*

Ключові слова: *розпаралелення, паралельні обчислення, GPU, SIMD, дискретні динамічні моделі.*

P. Stakhiv, I. Strubytska, Y. Kozak

PARALLELIZATION OF PROCESS OF BUILDING DISCRETE DYNAMICAL MODEL TWO-WINDING TRANSFORMER

The summary. *The process of building of discrete dynamic model of two-winding transformer was paralleled. Computing architecture SIMD was selected for parallelization. Graphic multi core processor was selected as devise of parallelization. The linear model of two-winding transformer was built for experimental set using program. For check accuracy of created model was conducted comparisons values, which was obtained using the model, with actual values. Numerical experiments, which confirmed the effectiveness of parallelization of this problem, were conducted. Namely, the execution time of problem at the central and graphics processor elements was compared.*

Key words: *parallelization, parallel computing, GPU, SIMD, discrete dynamical model.*

Постановка проблеми. Одним із перспективних підходів до моделювання є використання дискретних динамічних моделей, отриманих на основі оптимізаційного підходу. Головною перевагою даного підходу є його універсальність як стосовно класу об’єктів, що моделюються, так і математичної форми представлення результату. Також цей підхід легко піддається автоматизації, тому він є актуальним в умовах інтенсивного застосування обчислювальних методів, які орієнтовані на сучасні обчислювані засоби. Проте даний підхід володіє суттєвим недоліком – складністю отримуваної оптимізаційної задачі.

В даній роботі розглядається один із можливих напрямів, які дозволяють прискорити розв’язання оптимізаційної задачі, а саме, використання розпаралелення на основі SIMD архітектури. Засобом розпаралелення вибрано графічний процесор, який володіє значно кращим співвідношенням швидкість/ціна, ніж центральні процесори.

Аналіз останніх досліджень і публікацій. Задачі параметричної ідентифікації дискретних динамічних моделей достатньою мірою описані в працях Л. Заде та Ч. Дезоера [1], В. Стрейца [2]. З точки зору комп’ютерної імітації найперспективнішим є метод дискретних рівнянь стану [3–7]. Над стохастичними методами оптимізації працював Растригін Л. [8]. У працях Стахіва П.Г., Козака Ю.Я. [9–10] запропоновано використовувати оптимізацію для ідентифікації параметрів дискретних динамічних моделей. Проте такий підхід має високу обчислювальну складність, відповідно великими будуть і затрати машинного часу на побудову моделі. Використання графічних акселераторів для неграфічних обчислень є новим напрямком досліджень. Особливості застосування GPU і технології CUDA описано у працях [11–14]

Мета роботи. За допомогою паралельної програми для експериментального набору побудувати лінійну модель двообмоткового трансформатора вихідного каскаду підсилювача звукової частоти з осердям із пермалою. Для перевірки точності створеної моделі провести порівняння значень, отриманих за допомогою моделі, з фактичними значеннями. Для підтвердження ефективність розпаралелення даної задачі необхідно провести чисельні експерименти, а саме, порівняти час виконання задачі на центральному і графічному процесорних елементах.

Постановка задачі. Розглянемо узагальнену математичну модель у формі рівнянь стану (1), тому що дискретні значення зручніші для комп'ютерного моделювання:

$$\begin{cases} \vec{x}^{(k+1)} = F \cdot \vec{x}^{(k)} + G \cdot \vec{v}^{(k)} + \Phi(\vec{x}^{(k)}, \vec{v}^{(k)}) \\ \vec{y}^{(k+1)} = C \cdot \vec{x}^{(k+1)} + D \cdot \vec{v}^{(k)} \end{cases}, \tag{1}$$

де $\vec{x}^{(k)}$ – вектор змінних стану, який характеризує поточний стан об'єкта; $\vec{v}^{(k)}$ – вектор вхідних значень; $\vec{y}^{(k)}$ – вектор вихідних значень; F, G, C, D – матриці з невідомими коефіцієнтами, які необхідно знайти при побудові моделі; $\Phi(\vec{x}^{(k)}, \vec{v}^{(k)})$ – деяка нелінійна вектор-функція багатьох змінних, форму і коефіцієнти якої також потрібно знайти.

Така форма моделі (1) характеризується деяким вектором невідомих параметрів $\vec{\lambda}$. Для даної моделі цей вектор складається з елементів матриць F, G, C, D і коефіцієнтів вектор-функції $\Phi(\vec{x}^{(k)}, \vec{v}^{(k)})$.

Також введемо деякий критерій для точності моделі $Q(\vec{\lambda}) > 0$, який позначає відхилення поведінки моделі від поведінки модельованого об'єкта для відомих проміжків часу. Функція $Q(\vec{\lambda})$, яка називається функцією мети, розраховується як середньоквадратична похибка значень моделі від значень модельованої системи:

$$Q(\vec{\lambda}) > \sum_{k} \left| \vec{y} - \vec{y}' \right|^2$$

де $\vec{y}$ – характеристики, розраховані за допомогою моделі; $\vec{y}'$ – відомі характеристики об'єкта.

Тому побудову моделі можна звести до знаходження значень вектора $\vec{\lambda}$, при яких функція мети буде мінімальною. Ця задача розв'язується за допомогою алгоритму оптимізації.

Задача знаходження мінімальної точки нелінійної функції $Q(\vec{\lambda})$ багатьох змінних є складним завданням. При побудові дискретних динамічних моделей функція мети має чітко виражений яровий характер з великою кількістю локальних мінімумів. Для розв'язку таких задач найкращими характеристиками володіє метод напрямного конуса Растригіна [8]. За допомогою цього підходу можна провести цілеспрямований перебір локальних мінімумів, що прискорює знаходження глобального мінімуму цільової функції. Але обчислювальна складність такої задачі буде досить велика. Також для побудови якісної моделі використовується значна кількість апріорних даних. Отже, значними будуть і затрати машинного часу на реалізацію оптимізаційних процедур.

У праці [15] запропоновано використати розпаралелення даної задачі з використанням SIMD-архітектури.

Результати дослідження. Враховуючи, що в даному алгоритмі виконується велика кількість операцій над векторами, для практичної реалізації алгоритму розглянуто архітектуру процесорів SIMD. Даний тип архітектури дає змогу виконати один і той самий потік команд для багатьох потоків даних, що характерне для цієї задачі. Системи такого типу будуються таким чином, що процесорні елементи, які входять у систему, ідентичні, і всі вони керуються однією й тією ж послідовністю команд. Проте кожен процесор опрацьовує свій потік даних.

З практично доступних насьогодні пристроїв з SIMD-архітектурою та за критерієм ціна/продуктивність є GPU (Graphics Proccesing Unit) [14]. Тому пропонований алгоритм пошуку екстремуму потрібно адаптувати саме для цих акселераторів обчислень.

З програмного погляду реалізація такого розпаралелення не вимагає великих затрат завдяки програмній архітектурі графічного процесора. Для всіх потоків пересилається ідентичний програмний код. Вхідними даними для кожного потоку є параметри моделі, тобто вектор $\vec{\lambda}$. На виході кожного потоку отримують значення функції мети, яке обчислене у випадковій точці гіперконуса (для кожного потоку – своє).

Для програмної реалізації розпаралеленого алгоритму використано технологію CUDA (Compute Unified Device Architecture). На даний час обчислення на графічних процесорах з технологією CUDA – це інноваційне поєднання обчислювальних особливостей нового покоління графічних процесорів NVIDIA, що опрацьовують одразу тисячі потоків з високим рівнем інформаційного завантаження, які доступні через стандартну мову програмування С.

Для тестування розробленої програми побудуємо дискретну динамічну модель двообмоткового трансформатора вихідного каскаду підсилювача звукової частоти з осердям із пермалою. Вхідними величини є напруги на первинній і вторинній обмотках, а вихідними – відповідні сили струму. Перехідні характеристики трансформатора було знято експериментально. Спершу вимірювали сили струму в обох обмотках при поданні на первинну обмотку стрибка напруги при закороченій вторинній обмотці. Тоді вимірювали сили струму при стрибку напруги на вторинній і закороченій первинній обмотках. Частота дискретизації вимірювань складала 8 кГц.

Для існуючого набору експериментальних даних було побудовано лінійну неавтономну модель двообмоткового трансформатора

$$\begin{cases} \vec{x}^{(k+1)} = \begin{pmatrix} 0,9316 & 0,4713 \\ 0,0053 & 0.9343 \end{pmatrix} \cdot \vec{x}^{(k)} + \begin{pmatrix} 0,0111 & 0,0101 \\ -0,0011 & 0,0008 \end{pmatrix} \cdot \vec{v}^{(k)} \\ \vec{y}^{(k+1)} = \begin{pmatrix} 0,0018 & 0,0171 \\ 0,0011 & -0,0125 \end{pmatrix} \cdot \vec{x}^{(k+1)} + \begin{pmatrix} 0,0128 & 0,0008 \\ 0,0004 & 0,0078 \end{pmatrix} \cdot \vec{v}^{(k)} \end{cases}.$$

Фактичні дані сил струму на обмотках трансформатора у порівнянні з даними, які отримані за допомогою побудованої моделі, наведені на рисунках 1–4.

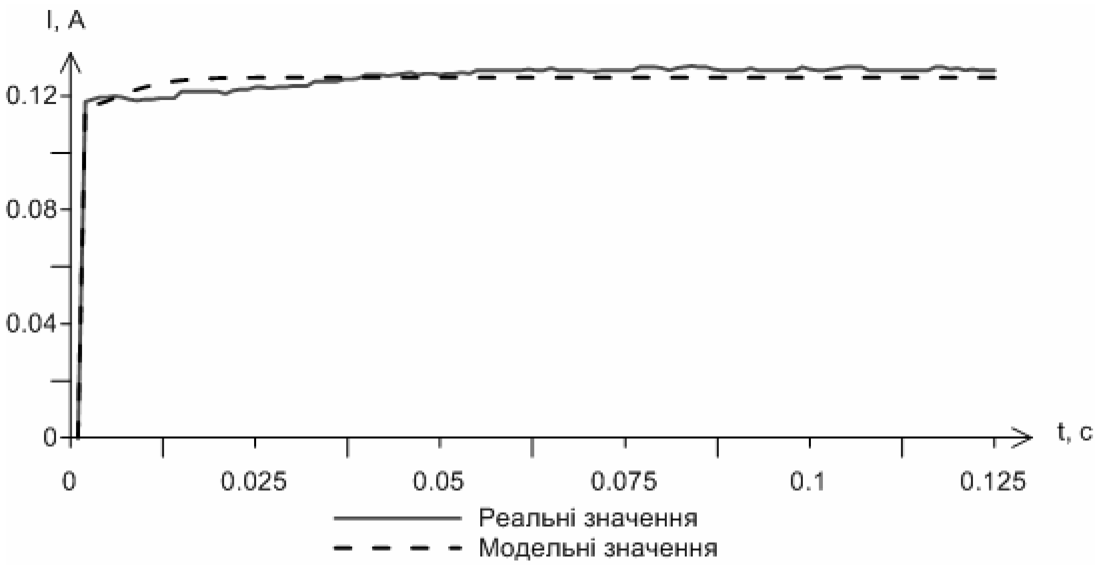

Рисунок 1. Залежність сили струму в первинній обмотці від часу при поданні сходинки напруги на первинну обмотку при закороченій вторинній обмотці

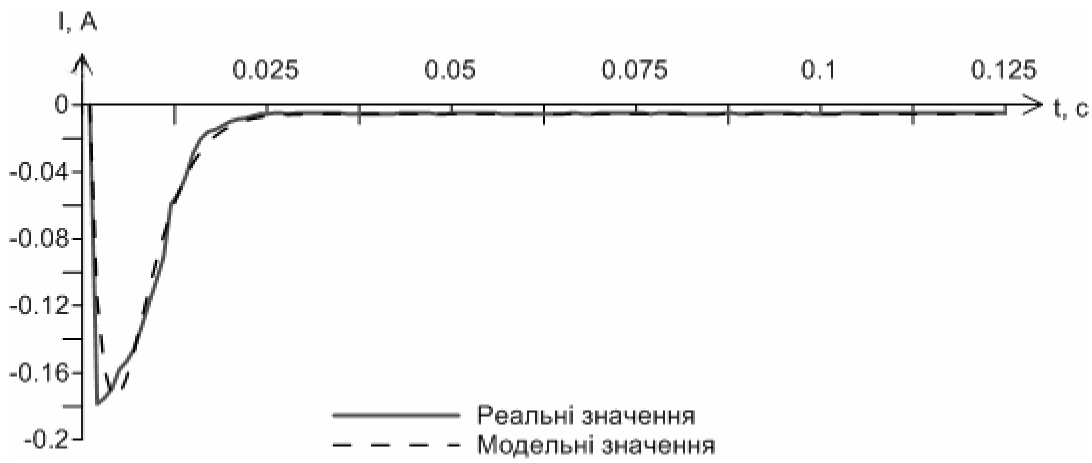

Рисунок 2. Залежність сили струму у вторинній обмотці від часу при поданні сходинки напруги на первинну обмотку при закороченій вторинній обмотці

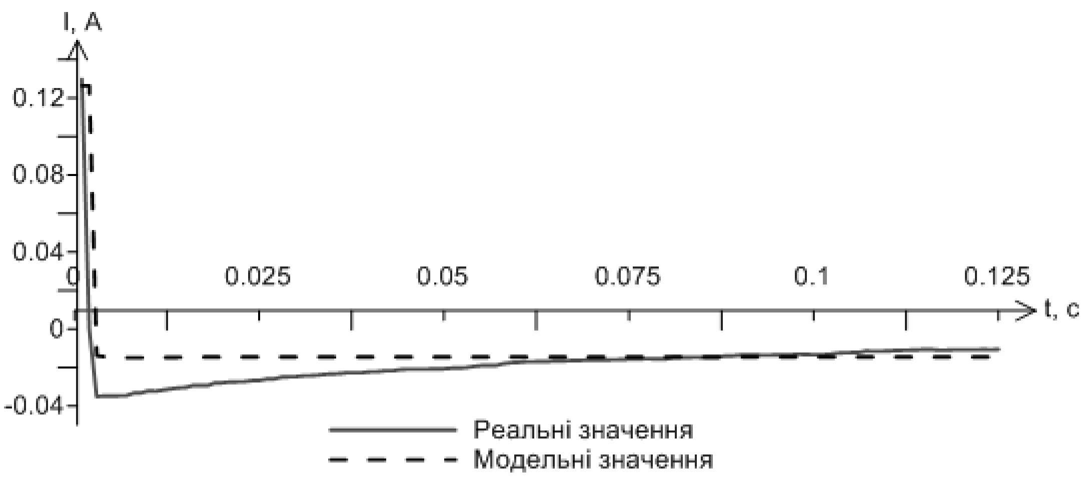

Рисунок 3. Залежність сили струму в первинній обмотці від часу при поданні сходинки напруги на вторинну обмотку при закороченій первинній обмотці

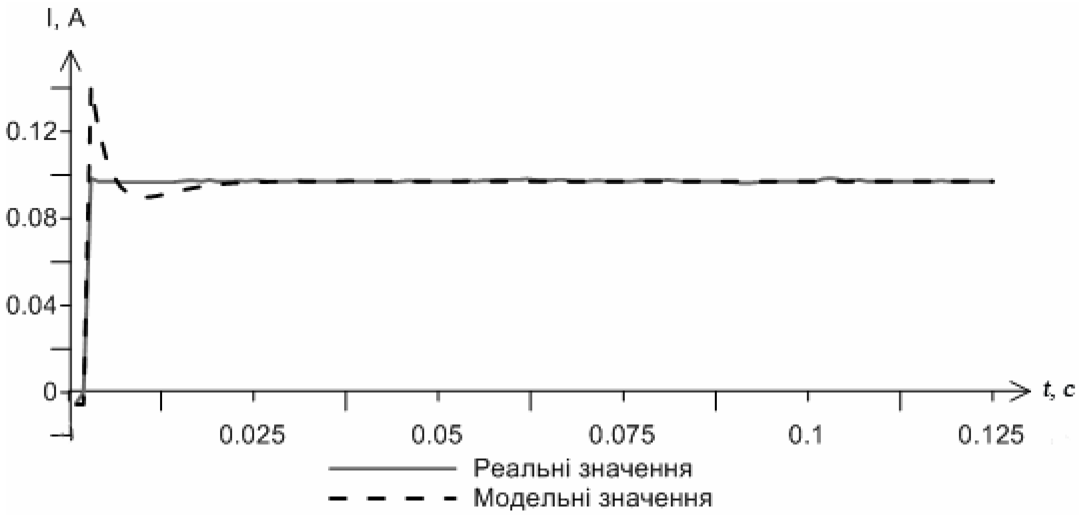

Рисунок 4. Залежність сили струму у вторинній обмотці від часу при поданні сходинки напруги на вторинну обмотку при закороченій первинній обмотці

Похибка побудованої моделі визначається за формулою

$$\varepsilon = \sqrt{\frac{\sum_{k=1}^{n}(y^{*(k)} - y^{(k)})^2}{\sum_{k=1}^{n}(y^{(k)})^2}} \cdot 100\%,$$

де $y^{(k)}$ – координата k-тої дискрети фактичного значення об’єкта; $y^{*(k)}$ – координата k-тої дискрети значення моделі; n – кількість дискрет.

Похибка прогнозування створеної моделі, обчисленої експериментальним шляхом, становить 7,05 %.

Досягнута ефективність розпаралелення залежить від кількості точок, для яких функція мети розраховується на один крок алгоритму оптимізації. Залежність часу виконання програми від кількості розрахунків функції мети для GPU (NVIDIA GeForce GTS250, 1024 Mb) і CPU (Core2Duo E8400, 3GHz) наведена на рис. 5.

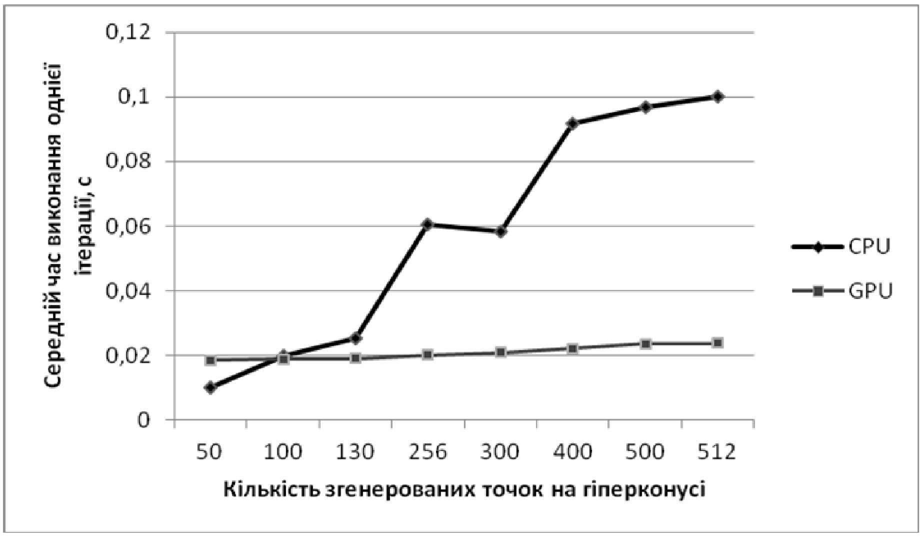

Рисунок 5. Залежність часу виконання програми від кількості розрахунків функції мети для GPU і CPU

Висновки. Реалізовано процедуру розпаралелення процесу побудови дискретних динамічних моделей за допомогою графічного процесора. Проведено чисельні експерименти, які показали, що паралельна програма на графічному процесорі виконується в 5 разів швидше за послідовну програму на CPU.

Список використаної літератури

1. Заде, Л. Теория линейных систем. Метод пространства состояний [Текст] / Л. Заде, Ч. Дезоер. – М.: Наука, 1970. – 704 с.
2. Стрейц В. Метод пространства состояний в теории дискретних линейных систем управления [Текст] / В. Стрейц; пер. с англ. под ред. Я.З. Цыпкина. – М.: Наука, 1985. – 296 с.
3. Стахив, П.Г. Анализ динамических режимов в электронных схемах с многополюсниками [Текст] / П.Г. Стахив. – Львов: Высш. школа, 1988. – 154 с.
4. Hinamoto T. Approximation of polynomial state-affine discrete-time systems / T. Hinamoto, S. Mackava – IEEE Trans. Circ. and Syst. – 1984. – Vol. 33, № 8. – P. 713–721.
5. Isidori A. Direct Construction of minimal bilinear realization from nonlinear input-output maps / A. Isidori – IEEE. – 1973. – Vol. AC-18, № 6. – P. 626–631.
6. Кунцевич, Р.М. О редуцированных моделях дискретных динамических объектов и их гарантированных оценках в задачах управления [Текст] / Р.М. Кунцевич // Проблемы управления и информатики. – 2001. – № 1. – С. 42–50.
7. Мельник, Б.К. Алгоритм побудови білінійних макромоделей багатополюсних елементів електронних схем [Текст] / Б.К. Мельник, П.Г. Стахів // Теоретична електротехніка. – 1995. – № 52. – С. 94–98.
8. Растригин, Л.А. Адаптация сложных систем [Текст] / Л.А. Растригин. – Рига: Зинатне, 1981. – 375 с.
9. Стахів, П.Г. Побудова макромоделей електромеханічних компонент із використанням оптимізації [Текст] / П.Г. Стахів, Ю.Я. Козак // Технічна електродинаміка. – 2001. – №4. – С. 33–36.
10. Стахів, П.Г. Побудова математичної макромоделі електромеханічного перетворювача вентильного двигуна з використанням оптимізації [Текст] / П.Г. Стахів, Ю.Я. Козак, В.Г. Гайдук // Електроенергетичні та електромеханічні системи // Вісник національного університету «Львівська політехніка». – 2001. – №418. – С. 159–164.
11. CUDA Zone – The resource for CUDA developer. C [Електронний ресурс]. – Режим доступу: http://www.nvidia.com/object/cuda_home_new.html.
12. Использование видеокарт для вычислений [Електронний ресурс]. – Режим доступа: http://gpgpu.ru.
13. NVIDIA CUDA Compute Unified Device Architecture, Programming Guide, Version 2.0. – 2008. –107 p.
14. Боресков, А.В. Основы работы с технологией CUDA [Текст] / А.В. Боресков, А.А. Харламов. – М.: ДМК Пресс, 2010. – 232 с.
15. Козак, Ю.Я. Розпаралелення алгоритму оптимізації параметрів дискретних динамічних моделей на масивно-паралельних процесорах [Текст] / Ю.Я., П.Г. Стахів, І.П. Струбицька // Відбір і обробка інформації. – 2010. – Вип. 32 (108). – С. 126–130.

Отримано 21.02.2012