ISSN 2224-087X. Електроніка та інформаційні технології. 2016. Випуск 6. С. 77–82
Electronics and information technologies. 2016. Issue 6. P. 77–82

УДК 519.622.2

ЗАСТОСУВАННЯ СТЕПЕНЕВОГО МЕТОДУ В АДАПТИВНОМУ АЛГОРИТМІ ІНТЕГРУВАННЯ ЖОРСТКИХ СИСТЕМ

Я. Кость

Львівський національний університет імені Івана Франка
вул. Ген. Тарнавського, 107, 79017 Львів, Україна
kostjerry@gmail.com

Виправлено недоліки в реалізації степеневого методу оцінювання найбільшого за модулем власного числа матриці загального вигляду, яка працює з комплексними власними числами. Надалі називатимемо виправлену реалізацію степеневого методу, що працює з комплексними власними числами, СМК. З'ясовано, що виправлений код працює ефективно для матриць з довільним набором власних чисел, які належать комплексній множині. Ефективність СМК досліджено в ході інтегрування моделі Ван-дер-Поля. Виконано порівняння швидкодії і точності СМК із однією з реалізацій методу QR. Виявлено, що СМК доцільно використовувати в адаптивному алгоритмі моделювання жорстких систем для зниження затрат обчислювальних ресурсів.

Ключові слова: степеневий метод, жорсткі системи, адаптивний алгоритм, найбільше за модулем власне число, комплексні власні числа.

Віддавна в основі кожної природничої науки є експеримент. Він підтверджує вірогідність теоретичних знань, а також визначає зв'язок із їхнім практичним застосуванням. Є два основні способи проведення експерименту: фізичний та числовий.

Часто числовий експеримент зводиться до розв'язування задачі Коші [1] інтегруванням системи диференціальних рівнянь (математичної моделі) на заданому проміжку із заданими початковими умовами.

У математичному моделюванні є особливий клас задач, розв'язування яких потребує значних затрат обчислювальних ресурсів. У цих задачах є швидкоплинні та повільні процеси. Швидкоплинні процеси (зазвичай, це перехідні процеси) накладають значні обмеження на крок інтегрування. Після їхнього загасання, коли домінують повільні процеси, крок інтегрування теоретично можна було б збільшувати. Проте незавершені швидкоплинні процеси, які вже не роблять видимого внеску в розв'язок, впливають на стійкість процесу інтегрування. Тобто якщо проміжки, що повільно змінюються, інтегрувати за допомогою явних методів та пробувати збільшувати крок інтегрування, то процес стає нестійким, розв'язки нагромаджують великі похибки і стають невірогідними. Отже, ділянки розв'язку, які повільно змінюються, потребують від явних методів невиправдано, з погляду задовільної точності, малого кроку. Такі задачі називають жорсткими [1].

ISSN 2224-087X. Електроніка та інформаційні технології. 2016. Випуск 6

Для розв'язування жорстких задач є спеціальний клас методів – неявні методи, які, проте, не є ефективними для нежорстких випадків. Цікаво, що жорсткість моделі може змінюватися в процесі її інтегрування. Тому часто намагаються обирати клас методу інтегрування динамічно. З літератури відомо [2, 3], що для цього вибору корисним є визначення найбільшого за модулем власного числа матриці Якобі лінеаризованої системи диференціальних рівнянь, яку інтегруємо. Оскільки в процесі інтегрування такі обчислення необхідно проводити багаторазово, то метод знаходження найбільшого за модулем власного числа матриці має бути швидким.

Також зазначимо, що цей метод не мусить бути особливо точним, оскільки для перемикання методів інтегрування нам достатньо наближеної оцінки максимального власного числа. Після аналізу літературних джерел ми вибрали ітераційні методи. Найшвидшим з них є степеневий.

Виправлена реалізація степеневого методу, що працює з комплексними власними числами (СМК). У найпростішому вигляді степеневий метод формулюють так:

$$y^{(k+1)} = Ay^{(k)}, \lambda_{\max}^{(k+1)} = \frac{y_j^{(k+1)}}{y_j^{(k)}}, \quad (1)$$

де $y^{(k)}$ – k-те наближення власного вектора; $y_j^{(k+1)}$ і $y_j^{(k)}$ – відповідні компоненти векторів $y^{(k+1)}$ та $y^{(k)}$ (у цьому разі як номер j можна використовувати будь-яке число з діапазону $j = \overline{1,n}$); n – розмір вектора y.

До переваг методу належить те, що він є простий у реалізації, ітераційний та швидкий. Проте метод має і недоліки, а саме:

- він повільно збігається, коли два найбільші за модулем власні числа є близькими за значенням;
- потрібно задавати початкове наближення власного вектора, алгоритм оптимального вибору якого чітко не описаний у літературі;
- стандартний алгоритм степеневого методу не працює, коли найбільше за модулем власне число є комплексним.

Ми дослідили код програми степеневого методу на C++ [4], який працює для випадків, коли матриця має комплексно-спряжені власні числа. На жаль, доступ до першоджерела [5] алгоритму закритий для вільного використання, тому довелось аналізувати код його програмної реалізації. Програма на кожній ітерації обчислює три послідовні наближення власного вектора і певним чином їх нормує. Після цього знайдені наближення власного вектора формують коефіцієнти квадратного рівняння (2). Коренями цього рівняння є власні числа матриці, одне з яких найбільше за модулем серед усіх власних чисел матриці:

$$\lambda^2 + \alpha\lambda + \beta = 0. \quad (2)$$

Тестування цього програмного коду засвідчило, що він має недолік. Алгоритм не збігався для деяких випадків, коли корені квадратного рівняння були дійсними. Після аналізу коду виявлено логічне недоопрацювання. Результат повертався лише тоді, коли дискримінант був від'ємний. У такому випадку ми отримували два комплексно спряжені власні числа. Додатний дискримінант рівняння (2) зациклював програму так, що вона не повертала результату.

Ми усунули згаданий недолік у програмному коді. Також ми видалили зайві обчислення власного вектора, оскільки для оцінки жорсткості нам необхідне лише найбільше за модулем власне число матриці. Програмний код переписано мовою програмування C# та інтегровано у нашу систему моделювання жорстких динамічних систем [6].

Тестування методу СМК. Тестові задачі обирали такими, щоб охопити характерні випадки власних чисел, які, на наш погляд, могли б вплинути на ефективність методу.

Тест 1. $\begin{pmatrix} 4 & 3 \\ -2 & -3 \end{pmatrix}$ $\lambda_1 = -2$, $\lambda_2 = 3$. Тест 2. $\begin{pmatrix} 5 & 17 \\ -1 & -3 \end{pmatrix}$ $\lambda_1 = 1 + i$, $\lambda_2 = 1 - i$.

Тест 3. $\begin{pmatrix} 5 & 10 \\ -1 & -2 \end{pmatrix}$ $\lambda_1 = 0$, $\lambda_2 = 3$. Тест 4. $\begin{pmatrix} 5 & 15{,}996 \\ -1 & -2{,}999 \end{pmatrix}$ $\lambda_1 = 1$, $\lambda_2 = 1{,}001$.

Тест 5. $\begin{pmatrix} 2 & 1 \\ -1 & 4 \end{pmatrix}$ $\lambda_1 = 3$, $\lambda_2 = 3$. Тест 6. $\begin{pmatrix} 5 & -3 \\ 7 & -5 \end{pmatrix}$ $\lambda_1 = -2$, $\lambda_2 = 2$.

Тест 7. $\begin{pmatrix} 5 & 17 & 4 \\ -1 & -3 & 2 \\ 8 & 11 & 1 \end{pmatrix}$ $\lambda_1 \approx 10{,}08$, $\lambda_2 \approx -3{,}54 + 2{,}98i$, $\lambda_3 \approx -3{,}54 - 2{,}98i$.

Тест 8. $\begin{pmatrix} 1 & 30 & 4 \\ -10 & -3 & 2 \\ 8 & 25 & 1 \end{pmatrix}$ $\lambda_1 \approx -0{,}08 + 14{,}59i$, $\lambda_2 \approx -0{,}08 - 14{,}59i$, $\lambda_3 \approx -0{,}83$.

Ми порівняли швидкодію і точність СМК із готовою реалізацією [7] методу QR [8]. Метод QR узято за основу для порівняння тому, що він знаходить усі власні числа матриці загального вигляду та є одним із найліпших для розв'язування повної класичної матричної проблеми. Для нашої реалізації СМК задано похибку $\varepsilon = 10^{-2}$.

Порівняння швидкодії та точності СМК із QR

Номер тесту	Кількість ітерацій	Коефіцієнт зростання швидкодії	Відносна похибка $\lvert\lambda_{max}\rvert$, %
1	1	4,2	0,00
2	1	2,0	0,00
3	0	5,9	0,00
4	1	2,5	0,00
5	1	4,2	0,00
6	1	4,7	0,00
7	4	1,9	0,05
8	2	3,6	0,10

З таблиці бачимо, що СМК дає значний виграш у швидкодії порівняно з розглянутою реалізацією QR, зберігаючи розв'язки в межах допустимої похибки.

Використання СМК у процесі інтегрування жорстких систем. Для дослідження роботи СМК у нашій системі моделювання взято тестову модель Ван-дер-Поля (3) та проінтегровано її методом середньої точки [9] з кроком $h = 10^{-2}$.

$$\ddot{x} - \mu(1 - x^2(t))\dot{x} + x(t) = 0, \quad \mu = 10, \quad t \in [0; 2], \quad x(t_0) = 1, \quad \dot{x}(t_0) = 0. \tag{3}$$

На кожному кроці було заморожено коефіцієнти моделі, лінеаризовано її та знайдено її найбільше за модулем власне число методами QR та СМК.

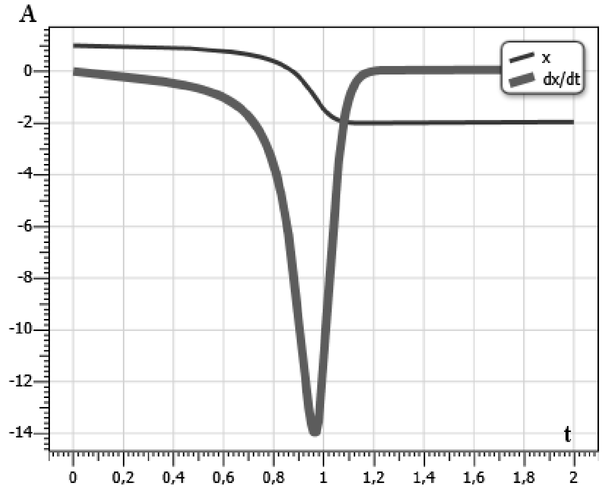

Рис. 1. Розв’язок моделі Ван-дер-Поля.

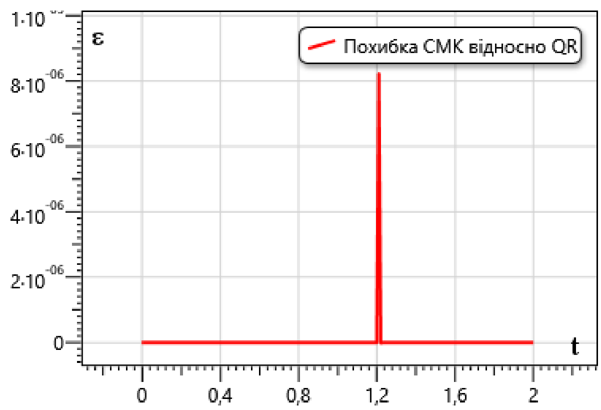

Рис. 2. Похибка СМК порівняно з QR.

З рис. 2 бачимо, що за точністю обчислень СМК майже не поступається QR.

На рис. 3 зображено виграш у швидкодії СМК відносно QR. На рис. 4 графік збільшено в околі 1. Там, де графік розміщений над пунктирною лінією, є виграш у швидкодії.

На жаль, ми не можемо пояснити піки на рис. 2 та 3. На рис 3 у разі повторення досліду вони не зберігають попереднього положення. Імовірно, піки коефіцієнта зростання швидкодії мають стохастичний характер.

Програму моделювання жорстких динамічних систем було налаштовано так, щоб автоматично отримувати графік динаміки найбільшого за модулем власного числа матриці Якобі у процесі інтегрування (рис. 5).

Якщо додати на цей графік області стійкості методів, то це дасть змогу візуально оцінювати оптимальний вибір методу інтегрування.

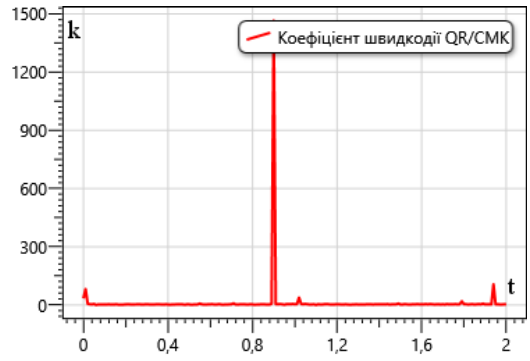

Рис. 3. Коефіцієнт зростання швидкодії СМК відносно QR.

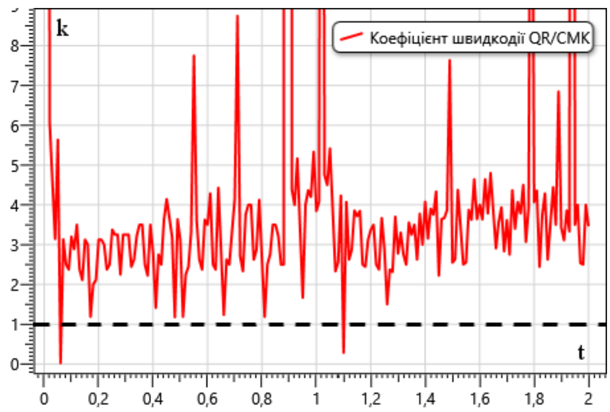

Рис. 4. Коефіцієнт зростання швидкодії СМК відносно QR (фрагмент).

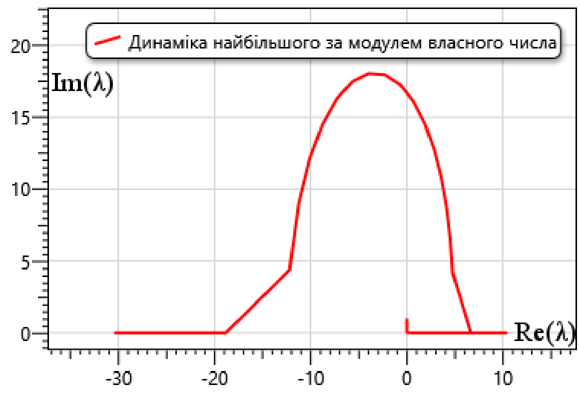

Рис. 5. Динаміка зміни найбільшого за модулем власного числа матриці Якобі на комплексній площині на заданому проміжку інтегрування.

З проведених дослідів зроблено висновок, що метод СМК працює швидше, ніж метод QR, зберігаючи задовільною точність обчислень.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. *Хвищун І. О*. Програмування і математичне моделювання : [підручник] / І. О. Хвищун. – К. : Ін Юре, 2007. – 544 с.
2. *Новиков Е. А.* Явные методы для жестких систем / Е. А. Новиков ; [под ред. А. Н. Горбань]. – Новосибирск : Наука, 1997. – 195 с.
3. *Хайрер Э.* Решения обыкновенных дифференциальных уравнений. Жесткие и дифференциально-алгебраические задачи / Э. Хайрер, Г. Ваннер. – М. : Мир, 1999. – 685 с.
4. Реалізація модифікованого степеневого методу мовою програмування C++ [Електронний ресурс]. – Режим доступу : http://people.sc.fsu.edu/~jburkardt/cpp_src/power_method/power_method.cpp
5. VanDeVelde Eric. Concurrent Scientific Programming / E. VanDeVelde. – Springer, 1994, ISBN 0-387-94195-9, LC: QA76.58.V35.
6. *Кость Я. І.* Розробка програмного забезпечення для моделювання динамічних систем із жорсткими математичними моделями / Я. І. Кость, І. О. Хвищун // Електроніка та інформ. технології. – 2012. – Вип. 2. – С. 184–196.
7. Кросплатформна бібліотека для чисельного аналізу та обробки даних [Електронний ресурс]. – Режим доступу : http://www.alglib.net
8. *Уилкинсон Дж. Х.* Алгебраическая проблема собственных значений / Дж. Х. Уилкинсон. – М. : Наука, 1970. – 564 с.
9. *Кость Я. І.* Моделювання консервативних систем традиційними чисельними методами / Я. І. Кость, І. О. Хвищун, Я. А. Шмигельський // Електроніка та інформ. технології. – 2015. – Вип. 5. – С. 132–136.

Стаття: надійшла до редакції 12.05.2016,
доопрацьована 19.05.2016,
прийнята до друку 24.05.2016.

POWER METHOD IN THE ADAPTIVE SOLVER FOR STIFF SYSTEMS

Y. Kost

Ivan Franko National University of Lviv,
107 Tarnavsky St., UA–79017 Lviv, Ukraine
kostjerry@gmail.com

A bug fixed in the power method's program code (PMC) of evaluation of the largest in magnitude eigenvalue, that works with complex eigenvalues. Shown by the examples that PMC is effective for all matrices with arbitrary set of eigenvalues of the complex plain. The effectiveness of PMC in the integration process tested in the Van-der-Paul model. Speed and accuracy ratio of PMC comparing to the QR method code discussed. Shown that PMC is well suited for the use in adaptive solvers for stiff systems. This allows optimizing computing resources in the problems of mathematical modeling.

Key words: power method, stiff systems, adaptive solver, maximal eigenvalue, complex eigenvalues.