УДК 517.956 (043.2)

О.М. Глазок, кандидат технічних наук, Д.Ю. Катюха,
(Національний авіаційний університет, Україна)

ДОСЛІДЖЕННЯ ШВИДКОСТІ ВИКОНАННЯ ОПЕРАЦІЙ ПІДВИЩЕНОЇ РОЗРЯДНОСТІ В ХОДІ ІТЕРАЦІЙНОГО ПОШУКУ РОЗВ'ЯЗКУ СИСТЕМИ ЛІНІЙНИХ РІВНЯНЬ

Розглядається ітераційний метод розв'язання систем лінійних алгебраїчних рівнянь, побудований на основі другого методу Ляпунова. Досліджено залежність часу обчислень при використанні арифметичних операцій з підвищеною точністю від застосованої розрядності розрахунків. Використано алгоритм довгої арифметики, реалізований у вигляді класу C++ для обробки довгих чисел

Існує велика кількість наукових та прикладних задач, що зводяться до чисельного розв'язання систем лінійних алгебраїчних рівнянь. Теорія отримання точних та наближених розв'язків систем лінійних алгебраїчних рівнянь є досить старою і дослідженою галуззю обчислювальної математики. Існує обширна література, присвячена методам прикладної лінійної алгебри, а програмні продукти, що реалізують найбільш популярні алгоритми обчислювальної лінійної алгебри, стали невід'ємною частиною прикладного програмного забезпечення, зокрема, сучасних математичних пакетів.

Сучасний бурхливий розвиток і розповсюдження обчислювальної техніки надає в розпорядження дослідника обчислювальні потужності, що дозволяють розв'язувати задачі великих розмірностей; в той же час в математичному забезпеченні спостерігається ряд проблем, що не дозволяють ефективно використати ці можливості. Серед труднощів практичного розв'язання систем лінійних алгебраїчних рівнянь великої розмірності найперше слід відзначити обмеженість оперативної пам'яті ЕОМ та значну тривалість розрахунків. Розмірність задач, а відтак і обсяг даних, що необхідно зберігати та обробити при їх розв'язанні, зростають швидше, ніж обсяг оперативної пам'яті доступних обчислювальних машин, і швидше, ніж їх швидкісні характеристики. В значній мірі обмеження на розмірність систем можна б було зняти, якщо використовувати для зберігання елементів матриці зовнішні запам'ятовуючі пристрої. Однак швидкість обміну даними з ними невисока, тому в цьому випадку суттєво зростають як витрати машинного часу, так і складність відповідних алгоритмів.

Алгоритми, реалізовані в згаданому вище програмному забезпеченні, як правило, відносяться до класу алгоритмів прямого розв'язання задачі, і орієнтовані на розв'язання систем лінійних алгебраїчних рівнянь невеликих чи середніх розмірів (до кількох сотень чи тисяч рівнянь), в той час як сучасні прикладні задачі в результаті апроксимації неперервного функційного рівняння кінцево-різницевою задачею нерідко породжують системи, у яких кількість рівнянь може складати сотні тисяч чи навіть мільони, і потреби практики в розв'язанні задач все більшої розмірності зростають.

Вказані протиріччя мотивують дослідників пропонувати нові методи розв'язання таких задач. Так, для ряду часткових випадків систем з розрідженою матрицею було створено економні прямі методи розв'язання [1, 2]. Було також розроблено ряд ітеративних методів; зокрема, суттєвим кроком вперед стали метод підпросторів Крилова та інші подібні градієнтно-орієнтовані методи, що використовують спряжені підпростори [3, 4]. Однак залишається і ряд невирішених проблем. Зокрема, більшість запропонованих методів розв'язання таких систем розроблено в припущенні, що матриця системи має стрічкову структуру, причому ширина стрічки значно менша порядку матриці. Однак це припущення виконується не для всіх задач. Крім того, запропоновані методи дозволяють збільшити порядок розв'язуваної системи лише за умови відповідного (за порядком величин) нарощування обсягу ресурсів, що мають бути використані обчислювальною системою для її розв'язання.

Ітераційні методи застосовують головним чином для розв'язання задач великої розмірності, коли використання прямих методів неможливе через названі вище обмеження. Не вдається до таких задач застосувати і методи з виключенням, оскільки при їх використанні велике число нульових елементів перетворюється на ненульові і матриця втрачає властивість розрідженості. Важливою перевагою ітераційних методів є те, що в ході ітераційного процесу матриця не міняється, і вона, природно, залишається розрідженою. Велика ефективність ітераційних методів в порівнянні з прямими методами тісно пов'язана з можливістю істотного використання розрідженості матриць. Можна зробити висновок, що актуальним є пошук нових ітераційних методів розв'язання названих задач, орієнтованих на роботу з системами рівнянь великої розмірності. Перспективним напрямком такого пошуку виглядає застосування методу функцій Ляпунова. Теорія першого та другого методів Ляпунова продовжує активно розвиватися, зокрема в застосуваннях до розв'язання задач, в яких описується чи визначається поведінка систем з розподіленими параметрами.

В системах з великою кількістю елементів зв'язки між елементами найчастіше є локальними, і, відповідно, матриці систем лінійних алгебраїчних рівнянь, що описують такі системи, найчастіше є розрідженими. Тому є можливість за рахунок раціональної організації обчислювального процесу добитися значного зменшення потреб у оперативній пам'яті для зберігання даних, оскільки немає потреби зберігати в пам'яті нульові елементи (яких у розрідженій матриці переважна більшість), а також зменшення тривалості обчислень – завдяки тому, що немає потреби виконувати над нульовими елементами арифметичні дії, результат яких заздалегідь відомий.

Оцінимо виграш, отриманий від відкидання нульових елементів для m-діагональної матриці розмірності N (так, наприклад, для класичної різницевої схеми, що наближує диференційне рівняння другого порядку щодо функції однієї координати, m=3, а N відповідає кількості вузлів сітки.) У головній діагоналі матриці (що містить ненульові елементи) міститься N елементів, а інші (m–1) діагоналей з ненульовими елементами містять на один чи декілька елементів менше. Оцінимо кількість елементів в цих діагоналях зверху також величиною N . Таким чином, всього у матриці є менше ніж $m\times N$ елементів, що не є завідомо нульовими. А відносний вміст ненульових елементів менший, ніж $\frac{mN}{N^2}=\frac{m}{N}$. Решта елементів – більше ніж $N(N–m)$ елементів – нульові, а відносний вміст нульових елементів більший, ніж $\frac{N-m}{N^2}$. Величина m відповідає кількості зв'язків одного елемента і в більшості задач, де зв'язки між елементами локальні, є невеликим числом. Розмірність задачі N може сягати сотень тисяч і мільонів, отже, видно, що раціоналізація обчислень приносить значну економію. До цього слід додати, що, якщо в задачі задано крайові значення, то і в «ненульових» діагоналях частина елементів можуть виявитись нульовими, оскільки крайові значення враховуються у постійному векторі правих частин.

При спробах використання непрямокутних сіток та сіток, геометрія яких не відповідає формі об'єктів, було відзначено такі проблеми [5]:

1. У випадку невідповідності між сіткою та границею кроки-«сходинки» на моделі границі приводять до зростання помилок в рішеннях, особливо коли сітка має великий крок. Для криволінійних границь уникнути такої невідповідності практично неможливо через фундаментальне протиріччя між неперервністю кривих, що утворюють границю, та дискретністю сітки.

2. Складність реалізації алгоритмів розрахунку. Кількість вузлів (точок) сітки у одному рядку не постійна, і це вимагає або застосування непрямої адресації, або введення додаткового масиву, який обмежує діапазон індексів в кожному рядку. При цьому код програми, імовірно, доведеться змінювати для кожної нової задачі.

3. Урахування граничних умов вимагає особливої уваги.

Наступним кроком до зменшення апаратних вимог є застосування ітераційних методів розв'язання замість прямих. Розглянемо задачу пошуку розв'язку системи лінійних алгебраїчних рівнянь, записаної у векторно-матричній формі як

$$AX=B \quad , \tag{1}$$

де A – задана постійна матриця, B – заданий постійний вектор правих частин рівнянь, X – вектор розв'язків, який необхідно знайти.

Щоб застосувати другий метод Ляпунова, визначимо скалярну допоміжну функцію

$$V(X) = (AX–B)^T Q(AX–B) \quad , \tag{2}$$

де Q – симетрична вагова матриця. Розглянемо вектор X як вектор-функцію часу $X(t)$, і організуємо ітераційний процес так, щоб значення сходилися до точного розв'язку системи (1). Нехай закон загасания допоміжної функції (2) в часі задано рівнянням

$$\dot{V} + cV = 0 \quad , \tag{3}$$

де c – деяка постійна величина, значення якої можемо обрати довільно, таким чином визначаючи швидкість спадання норми нев'язки. Нехай також задано закон руху X :

$$\dot{X} = -k\frac{\partial V}{\partial X} \quad , \tag{4}$$

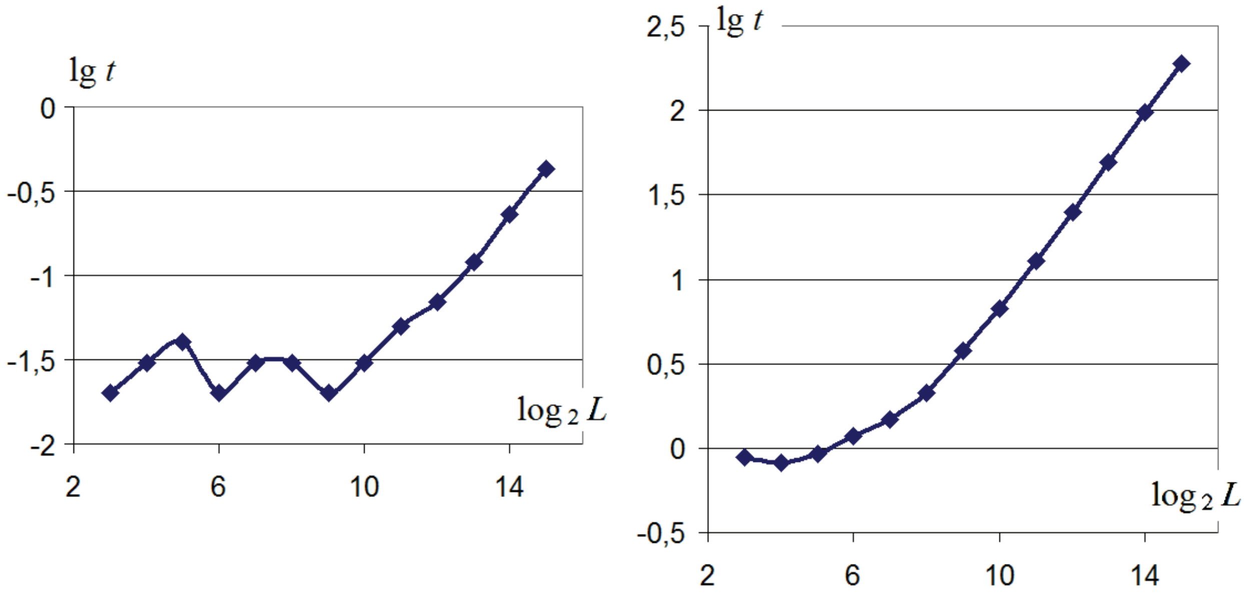
де k – деяка постійна (не довільна) величина.

Використовуючи визначення (2), (3) та визначаючи коефіцієнт k за умовами задачі, отримаємо на основі (4) необхідне диференційне рівняння, у якому похідна $\frac{\partial V}{\partial X}$ задана в силу системи (1), а права частина не містить невідомих величин.

При розв'язанні прикладної задачі за таким методом є практично важливою проблема знаходження компромісу між розмірністю системи, діапазоном представлення величин і точністю обчислень. Вдалий1 або невдалий вибір такого компромісу відбивається на достовірності отриманих рішень. Недостатня точність обчислень призводить до накопичення помилок при виконанні обчислювальних операцій. На даний час загальноприйнятим стандартом для представлення чисел у форматі з плаваючою крапкою і виконання операцій над ними є стандарт IEEE 754 [6]. Однак в літературі неодноразово піднімалося питання про те, що використання сучасних підходів до представлення дійсних чисел приводить до нераціональних витрат обчислювальних ресурсів; недостатньої достовірності результатів складних математичних розрахунків, особливо з накопиченням помилки; поганої відповідності моделі представлення дійсних чисел реальним потребам різних галузей знань.

Зрозуміло, що не вдасться досягти адекватності математичної моделі в тому випадку, якщо не вдається контрольовати погрішності обчислень. У зв'язку з цим є актуальним пошук шляхів для визначення ефективних форматів машинного представлення значень, що виникають при розв'язку задачі, та ефективних способів виконання операцій над ними.

У виконаному дослідженні було проведено вимірювання часу виконання арифметичних операцій в процесі розв'язання системи (1) з використанням алгоритму довгої арифметики, що був реалізований у вигляді класу C++ для обробки довгих чисел. Приклади результатів вимірювання подано на рис. 1. На осях абсцис графіків відкладено логарифм довжини чисел, що оброблялися. Як можна побачити, в обох випадках при нарощуванні розрядності залежність часу обчислень від розрядності наближається до лінійної, в той час як на початковому етапі вимірювань (при невеликих розрядностях) на тривалість розрахунків справляють переважний вплив побічні, псевдовипадкові фактори, оскільки доля власне арифметичних операцій при цьому невисока.



а *б*

Рис. 1. Результати чисельного експерименту: *а* – з використанням бібліотечних функцій; *б* – з використанням методів класу Longa.

Висновки

На основі другого методу Ляпунова було запропоновано метод отримання розв'язку системи лінійних алгебраїчних рівнянь (1) як границі, до якої сходиться послідовність наближених розв'язків диференційного рівняння щодо вектор-функції X, яке не містить інших невідомих величин. Це рівняння можливо розв'язати шляхом чисельного інтегрування на ЕОМ за одним з відомих методів, обравши деяке початкове наближення вектора X. При цьому швидкість руху до розв'язку та точність отриманих результатів визначається розрядністю обчислень. Виконано вимірювання часу операцій над багаторозрядними числами, які є частиною ітераційного процесу розв'язку системи рівнянь.

Важливим напрямком подальших досліджень є пошук ефективних для даного обчислювального алгоритму способів машинної обробки дійсних чисел, а також встановлення залежності між розрядністю обробки проміжних даних (і відповідно витратою часу) і швидкістю сходимості процесу пошуку розв'язку, і з'ясування умов, на основі яких можна обрати задовільний компроміс між точністю та швидкістю.

Список літератури

1. *Воеводин В.В., Кузнецов Ю.А.* Матрицы и вычисления. – М.:Наука, 1984. – 320 с.

2. *Джордж А., Лю Дж.* Численное решение больших разреженных систем уравнений: Пер. с англ. – М.: Мир, 1984. – 333 с.

3. *Петров И.Б., Лобанов А.И.* Лекции по вычислительной математике. – М.: БИНОМ. Лаборатория знаний, 2006. – 523 с.

4. *Ерёмин М.А.* Определитель Ерёмина в линейной и нелинейной алгебре. Линейное и нелинейное программирование. – М: КомКнига, 2006. – 120 стр.

5. *Ferziger J.H., Peric M.* Computational Methods for Fluid Dynamics. – Springer, 2002. – 423 p.

6. IEEE Standard for Floating-Point Arithmetic (Revision of IEEE Std 754-1985) // IEEE Computer Society. – 2008. – 70 p.