

О РАСПАРАЛЛЕЛИВАНИИ МЕТОДА ЧИСЛЕННОГО НАХОЖДЕНИЯ ЛЯПУНОВСКИХ ЭКСПОНЕНТ С ПОМОЩЬЮ OPENMP

Бурова И. Г., д.ф.-м.н, проф. СПбГУ
Мокаев Т. Н., к.ф.-м.н., мл. науч. сотр. СПбГУ
Федоров Е. Г., студент мат.-мех. факультета СПбГУ

Аннотация

В данной статье описан метод нахождения ляпуновских показателей с помощью алгоритма Бенеттина. А также сравнивается время вычисления при последовательной и параллельной реализациях множественного запуска данного алгоритма.

Введение

Ляпуновские показатели являются важными числовыми характеристиками динамической системы. Нередко необходимо вычислять эти показатели для большого количества траекторий, например при оценке ляпуновской размерности инвариантного множества [1, 2]. Из-за большой вычислительной сложности алгоритма, это может занять много времени. Для ускорения вычислений можно воспользоваться стандартом *OpenMP* для распараллеливания программы, распределив траектории по разным потокам.

Определение ляпуновских показателей

Дадим определение ляпуновских показателей в соответствии с работами [3, 4].

Пусть дана система дифференциальных уравнений

$$\frac{dx}{dt} = f(x), t \in \mathbb{R}, x \in U, \quad (1)$$

где U — открытое подмножество \mathbb{R}^n ($U \subseteq \mathbb{R}^n$), f непрерывно дифференцируемая вектор функция $U \rightarrow \mathbb{R}^n$. Пусть $\varphi(t, x_0)$ — решение этой системы такое, что $\varphi(0, x_0) = x_0$.

Рассмотрим линеаризацию системы (1) вдоль решения $\varphi(t, x_0)$:

$$\dot{y} = J(\varphi(t, x_0))y, \quad J(x) = Df(x), \quad (2)$$

где J — $n \times n$ матрица Якоби. Пусть $D(\varphi(t, x_0)) = (y_1(t, x_0), \dots, y_n(t, x_0))$ фундаментальная матрица линеаризованной системы (2) такая, что $D(\varphi(0, x_0)) = I$, где I — единичная $n \times n$ матрица. Теперь можно ввести следующее определение:

Определение 1. *Ляпуновским показателем $\mu_i(x_0)$ называется число*

$$\mu_i(x_0) = \overline{\lim}_{t \rightarrow +\infty} \frac{1}{t} \ln \|y_i(t, x_0)\|.$$

Показатель $\mu_i(x_0)$ называется строгим, если существует конечный предел

$$\mu_i(x_0) = \lim_{t \rightarrow +\infty} \frac{1}{t} \ln \|y_i(t, x_0)\|.$$

Алгоритм Бенеттина

Дадим краткое описание алгоритма Бенеттина [5, 6] для численного нахождения ляпуновских показателей траектории системы (1). Запишем вариационное уравнение системы (1) в следующем виде

$$\dot{X} = JX, \quad (3)$$

где X — $n \times n$ матрица, J — якобиан функции f .

Пусть x_0 — начальная точка траектории, $Q_0 = I$, где I — единичная $n \times n$ матрица. На i -ой итерации алгоритма исходная система (1) и её вариационное уравнение (3) интегрируются с начальными данными (x_{i-1}, Q_{i-1}) на временном интервале $(ih - h, ih)$, где h — малый интервал времени, параметр алгоритма. Обозначим результат интегрирования, как (x_i, H_i) . Затем матрица H_i QR -факторизуется: $H_i = Q_i R_i$. Для следующей итерации используется пара (x_i, Q_i) . Количество итераций N , также является параметром алгоритма.

В результате численную оценку j -ого ляпуновского показателя можно записать как

$$\mu_j(x_0) = \frac{1}{Nh} \sum_{i=1}^N \ln(R_i(j, j)) \quad (4)$$

Определение ляпуновских показателей и алгоритм их численного нахождения приведены только для направлений ортов декартовой системы, но также возможно обобщение для n линейно независимых векторов.

Сравнение последовательной и параллельной реализации

Для вычисления ляпуновских показателей были написаны последовательная и параллельная реализации на языке *C++*. Для распараллеленной реализации использовался открытый стандарт *OpenMP* (Open Multi-Processing). А именно, директивы *parallel* для создания потоков, директивы *for* для распределения вычислений по потокам, а также конструкции *shared* и *private* для задания типа переменных, используемых в параллельных секциях.

Пример части параллельной реализации:

```
#pragma omp parallel for shared(result_lces) num_threads(
    NUM_THREADS)
for (int iteration = 0; iteration < COUNT_POINTS; ++
    iteration)
{
    result_lces[iteration] = computeLCEs(system, dimension,
        POINTS[iteration], COUNT_ITERATIONS, D_TIME);
}
```

Обе реализации запускались с помощью компилятора из среды *Visual Studio 2013*. Операционная система *Windows 8.1*. Характеристики компьютера: 4-х ядерный процессор *Intel Core i5-5200U 2.20 GHz*.

Будем называть коэффициентом ускорения отношение времени работы распараллеленной реализации к последовательной. Теоретически коэффициент ускорения не может быть больше, чем $\frac{1}{\alpha + \frac{1-\alpha}{N}}$, где α доля расчетов, которые могут быть получены только последовательными вычислениями. На практике этот коэффициент еще меньше за счет накладных расходов на создание и содержание нескольких потоков вычисления, за счет окончательного формирования результатов по завершению вычислений во всех потоках. А также на коэффициент ускорения влияет загруженность процессора в данный момент задачами не относящимися к данной.

Для численного эксперимента была рассмотрена обобщенная система Лоренца [7, 8, 4, 9]:

$$\begin{cases} \dot{x} &= -\sigma(x - y) - \alpha yz \\ \dot{y} &= rx - y - xz \\ \dot{z} &= -bz + xy \end{cases}, \quad (5)$$

со значениями параметров $\sigma = 4, a = -0.66, r = 6.05, b = 1$.

Далее приведены четыре графика зависимости коэффициента ускорения от n — количества траекторий. Два графика соответствуют запуску на два потока, два других — на четыре потока. Параметры алгоритма: 1000 — количество итераций, 0.1 — временной шаг.

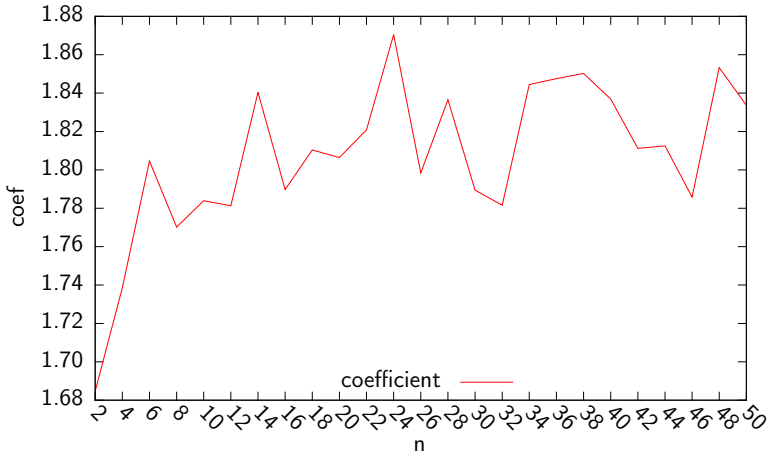


Рис. 1: Два потока, первый запуск

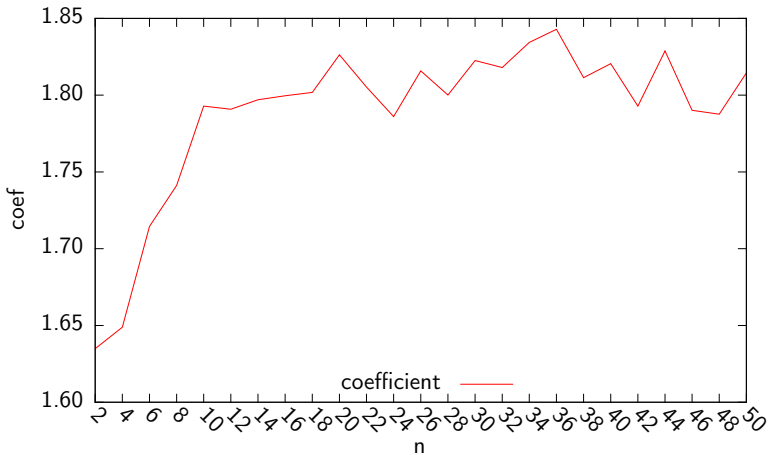


Рис. 2: Два потока, второй запуск

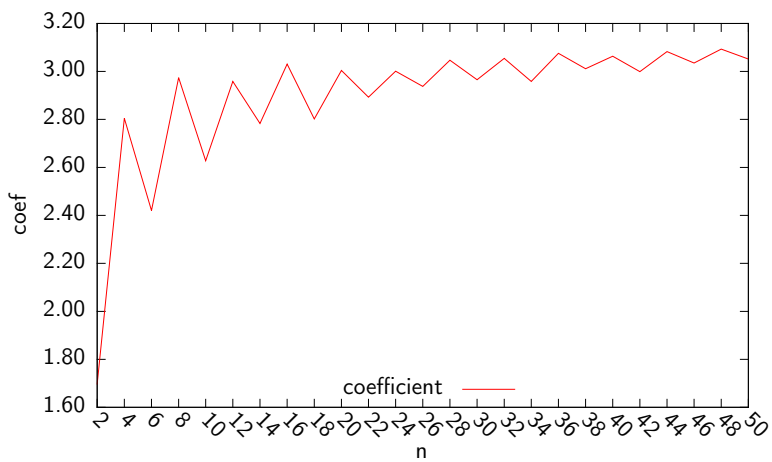


Рис. 3: Четыре потока, первый запуск

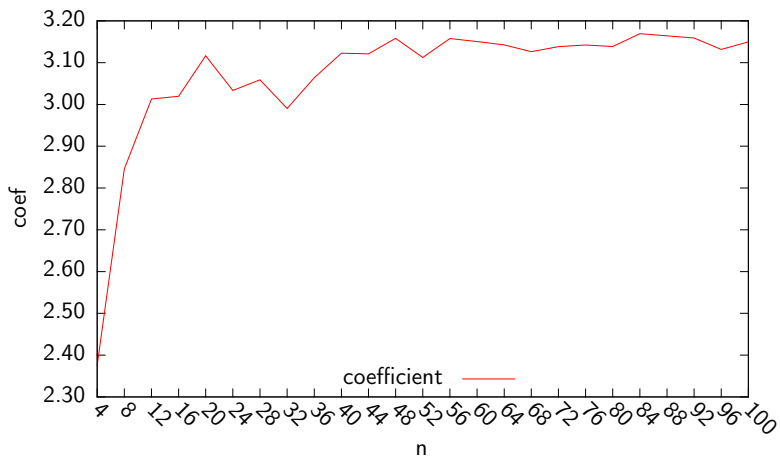


Рис. 4: Четыре потока, второй запуск

Для уменьшения влияния внешних факторов на коэффициент ускорения для каждого n последовательная и параллельная реализации запускались по очереди 100 раз. Коэффициент ускорения вычислялся как отношение суммарного времени работы всех запусков последовательной реализации к аналогичному показателю параллельной реализации.

Как видно из графиков, с ростом n растет и коэффициент ускорения, т.е. накладные расходы на создание и содержание потоков становятся менее значительными. Также видно, что при достаточно большом n параллельная реализация на четырех потоках работает в три раза быстрее, чем последовательная, что является значительным ускорением времени работы.

Заключение

В данной работе было дано определение ляпуновских показателей и описан алгоритм Бенеттина для их численного нахождения. Также показано, что с помощью *OpenMP* можно добиться значительного уменьшения времени работы программы.

Литература

- [1] *Леонов Г. А.* Странные аттракторы и классическая теория устойчивости движения. — Изд-во СПбГУ СПб., 2004.
- [2] *Kuznetsov N. V.* The Lyapunov dimension and its estimation via the Leonov method // *Physics Letters A.* — 2016.
- [3] *Lyapunov A. M.* The general problem of the stability of motion // *International Journal of Control.* — 1992. — Vol. 55, no. 3. — P. 531–534.
- [4] *Kuznetsov N. V., Mokaev T. N., Vasilyev P. A.* Numerical justification of Leonov conjecture on Lyapunov dimension of Rossler attractor // *Communications in Nonlinear Science and Numerical Simulation.* — 2014. — Vol. 19, no. 4. — P. 1027–1034.
- [5] Lyapunov characteristic exponents for smooth dynamical systems and for Hamiltonian systems; a method for computing all of them. Part 1: Theory / *G. Benettin, L. Galgani, A. Giorgilli, J.-M. Strelcyn* // *Meccanica.* — 1980. — Vol. 15, no. 1. — P. 9–20.

- [6] Lyapunov characteristic exponents for smooth dynamical systems and for Hamiltonian systems; a method for computing all of them. Part 2: Numerical application / *G. Benettin, L. Galgani, A. Giorgilli, J.-M. Strelcyn* // *Meccanica*. — 1980. — Vol. 15, no. 1. — P. 21–30.
- [7] *Рабинович М. И.* Стохастические автоколебания и турбулентность // *Успехи физических наук*. — 1978. — Т. 125, № 5. — С. 123–168.
- [8] *Leonov G. A., Boichenko V. A.* Lyapunov's direct method in the estimation of the Hausdorff dimension of attractors // *Acta Applicandae Mathematica*. — 1992. — Vol. 26, no. 1. — P. 1–60.
- [9] *Leonov G. A., Kuznetsov N. V., Mokaev T. N.* Homoclinic orbits, and self-excited and hidden attractors in a Lorenz-like system describing convective fluid motion // *The European Physical Journal Special Topics*. — 2015. — Vol. 224, no. 8. — P. 1421–1458.
- [10] *Антонов А. С.* Параллельное программирование с использованием технологии OpenMP: Учебное пособие. - Москва.: Изд-во МГУ, 2009. - 77 с.