

В. Н. Беловодский, М. Ю. Сухоруков

Донецкий национальный технический университет
belovodskiy@cs.dgtu.donetsk.ua, max.sukhorukov@gmail.com

СКАНИРУЮЩИЙ АЛГОРИТМ ПОСТРОЕНИЯ ОБЛАСТЕЙ ПРИТЯЖЕНИЯ ПЕРИОДИЧЕСКИХ РЕЖИМОВ

Рассмотрен алгоритм построения областей притяжения периодических режимов, описаны особенности его программной реализации. Приведены примеры качественного тестирования программы, выполнено сравнение с известными результатами.

Область притяжения, алгоритм построения, тестирование, MATLAB, C++

Введение

Задача определения областей притяжения стационарных режимов является одной из центральных в теории динамических систем [1] и к настоящему времени имеется ряд подходов в этом направлении. Так, в работе [2] Т. Хаяси рассматривает колебательные системы с одной степенью свободы, при выбранных соотношениях её параметров устанавливает весь спектр периодических режимов и выделяет неустойчивые из них. Далее, после перехода в исходной системе к «отрицательному» времени с использованием отображения Пуанкаре определяет сепаратрисы, проходящие через его неустойчивые неподвижные точки, которые, по существу, и делят фазовую плоскость на ряд зон, — областей притяжения. Этот подход получил дальнейшее развитие в работах А. Тондла [3], который методом медленно меняющихся амплитуд производит сведение исходных систем к автономному виду и выполняет классификацию неподвижных точек систем до четвертого порядка включительно. Характерной особенностью описанных методов является необходимость наличия достаточно полной информации о совокупности стационарных режимов движения, что в ряде случаев является весьма обременительным. Однако существуют методы и не налагающие этих требований. Один из них основан на теореме В. И. Зубова [4], устанавливающей необходимые и достаточные условия асимптотической устойчивости тривиального решения системы обыкновенных дифференциальных уравнений. Она является развитием прямого метода Ляпунова и построение области притяжения, в математическом плане, требует определения двух знакоопределенных функций, нахождение которых, в конечном счете, сводится к построению приближенных вычислительных процедур для решения уравнений в

частных производных [5]. Выигрывает, по сравнению с перечисленными, в части отсутствия ограничений на размерность задач и требований разработки специальных математических инструментов, метод Демиденко-Матвеевой [6]. Определение параметров, описывающих конфигурацию области асимптотической устойчивости тривиального решения, он дает в терминах норм матричного решения краевой задачи для дифференциального уравнения Ляпунова. Метод позволяет получить оценку области асимптотической устойчивости и его алгоритм вполне прозрачен. Вместе с тем, не умаляя достоинств ни одного из перечисленных подходов, следует обратить внимание на важность наличия такого программного продукта, который при относительной простоте был бы достаточно универсален, позволял, при желании, получать исчерпывающую информацию о конфигурации искомой области и который, также, можно было бы использовать в качестве контрольного или тестирующего при обработке более тонких и совершенных процедур. Таким мог бы стать метод, основанный на непосредственном варьировании начальных условий или, другими словами, — сканировании заданной части фазового пространства. Результатам разработки такого, далее, — сканирующего алгоритма и посвящена данная статья.

Алгоритм, его описание

Рассматриваемая задача заключается в следующем. Пусть имеется динамическая система

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t), \quad (1)$$

где $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, $\mathbf{f} = (f_1, f_2, \dots, f_n)^T$, полужирным шрифтом, здесь и далее, обозначается вектор-столбец. Необходимо в заданной части фазового пространства или его сечении $t = t_0$, выделить множество таких его точек, через которые проходят фазовые траектории в момент $t = t_0$ и, в дальнейшем, при $t > t_0$ приближаются к некоторому решению $\mathbf{x}^*(t)$ системы (1) периода T . Множество таких точек называется областью или бассейном притяжения решения $\mathbf{x}^*(t)$.

Концептуально обсуждаемый метод заключается в следующем. «Каждая» точка \mathbf{x}_0 , из заданной части D фазового пространства, последовательно выбирается в качестве начальной и для системы (1) решается задача Коши с начальным условием

$$\mathbf{x}(t_0) = \mathbf{x}_0. \quad (2)$$

При этом в фазовом пространстве системы (1) строится T -отображение Пуанкаре и формируется совокупность точек $\{\mathbf{x}(t_0 + mT)\}$, $m = 1, 2, 3, \dots$, которую в дальнейшем будем называть орбитой точки \mathbf{x}_0 . Как только, в некоторый момент элемент орбиты достаточно близко приближается к точке $\mathbf{x}^*(t_0) = \mathbf{x}_0^*$, — неподвижной точке отображения Пуанкаре, т.е. норма разности $\|\mathbf{x}(t_0 + mT) - \mathbf{x}_0^*\|$ становится меньше заданного ε , или же, орбита попадает в, уже построенный к этому моменту, фрагмент области притяжения решения $\mathbf{x}^*(t)$, то точка \mathbf{x}_0 также в него включается и помечается соответствующим образом. После пересмотра «всех» точек области D результат анализа выводится на экран.

Элементом, определяющим объём вычислений в данном алгоритме, является способ организации перебора точек области D . В базовой версии алгоритма он реализован так.

Первоначально формируется ядро области притяжения в виде шара $B_r(\mathbf{x}_0^*) = \{\mathbf{x} \in D : \|\mathbf{x} - \mathbf{x}_0^*\| < r\}$, принадлежащего области притяжения D_{x^*} решения $\mathbf{x}^*(t)$, с центром в точке \mathbf{x}_0^* наибольшего радиуса r . Для этого задаётся начальное значение радиуса $r = r_0$, производится последовательный перебор точек шара $B_{r_0}(\mathbf{x}_0^*)$ и анализ их орбит. Если $B_{r_0}(\mathbf{x}_0^*) \subset D_{x^*}$, то назначается следующее значение $r = r_1 > r_0$ и рассматриваются точки $\mathbf{x}_0 \in B_{r_1}(\mathbf{x}_0^*) \setminus B_{r_0}(\mathbf{x}_0^*)$. При этом дальнейшее построение орбиты прекращается, как только очередной её элемент попадает в $B_{r_0}(\mathbf{x}_0^*)$. Далее, производится изменение r с некоторым шагом Δr , т.е. полагается $r = r_2 > r_1$, $r = r_3 > r_2$ и т.д. до тех пор, пока в очередном шаре $B_{r_{k+1}}(\mathbf{x}_0^*)$ не обнаруживаются точки не принадлежащие D_{x^*} . В этом случае шар предыдущего радиуса $B_{r_k}(\mathbf{x}_0^*) = B_{x^*}$ и принимается в качестве ядра области притяжения.

Далее, сканирование области D путём анализа шаров $B_r(\mathbf{x}_0^*)$, последовательно увеличивающегося радиуса, продолжается. При этом, для каждой последующей точки, выбираемой в качестве начальной, процесс построения орбиты прекращается, как только очередной её элемент попадает в B_{x^*} .

Алгоритм предусматривает возможность одновременного построения областей притяжения нескольких стационарных режимов $\mathbf{x}_1^*(t), \dots, \mathbf{x}_k^*(t)$ периодов T_1, \dots, T_k , соответственно. В этом случае значение T назначается равным $\max_i T_i$, указанным выше образом строятся ядра

(шары) $B_{x_1}^*, \dots, B_{x_k}^*$ и, далее, без существенных отличий продолжается сканирование оставшейся части области D .

В дальнейшем, базовая версия алгоритма была развита в следующих направлениях. Вначале, с целью снижения объёма вычислений, построение орбиты прекращалось, как только очередной её элемент попадал в построенный к этому времени один из фрагментов области притяжения одного из режимов. Это осуществлялось с помощью специально формируемой для этих целей диагностической матрицы, размерность которой определяется числом уровней сканирования по каждой из фазовых переменных, а значения её элементов содержали информацию о типе режима. Далее, было снято, также, требование обязательного указания исследуемых стационарных режимов, точнее, им соответствующих неподвижных точек. В этом случае, на начальной стадии сканирования определяется первая из неподвижных точек x_1^* отображения Пуанкаре, строится ядро (шар) её области притяжения и, далее, продолжается сканирование области D указанным выше образом. Т.е. определяется, при её наличии, очередная неподвижная точка x_2^* и т.д. Обязательными исходными данными, в этом случае, являются лишь указание величины T , — максимального периода колебаний, и описание границ зоны сканирования D .

Особенности программного обеспечения

К настоящему времени разработаны два варианта программы, называемой BOA (Basin of Attractions), в среде MATLAB и на языке C++, реализующей описанный выше алгоритм. В каждом из них предусмотрены следующие возможности.

Пользователь описывает систему дифференциальных уравнений, задаёт область D , указывает, при желании, шаг сканирования s_D области D , а также, абсолютную ε_{abs} и относительную ε_{rel} погрешности интегрирования, общий период T изучаемых решений, предельное время интегрирования NT при расчете одного решения, шаг интегрирования Δt . По умолчанию значения некоторых из этих настроек равны: $s_D = 0.01$, $\varepsilon_{abs} = 10^{-6}$, $\varepsilon_{rel} = 10^{-3}$, $NT = 1000$, $\Delta t = 0.1$. В первом варианте программы пользователь выбирает также и вычислительную процедуру интегрирования из числа, имеющихся в среде MATLAB. Во втором варианте интегрирование осуществляется с помощью метода Рунге-Кутты 4-ого порядка.

В процессе выполнения программы автоматически производится оценка времени, необходимого для полного выполнения задачи. Это производится при предположении, что реализуемое, посредством

интегрирования системы (1), отображение Пуанкаре, является сжимающим и осуществляется следующим образом. По трём последовательным его точкам \mathbf{x}_k , \mathbf{x}_{k+1} , \mathbf{x}_{k+2} , где $T(\mathbf{x}_k) = \mathbf{x}_{k+1}$, $T(\mathbf{x}_{k+1}) = \mathbf{x}_{k+2}$, определяется предположительный коэффициент сжатия

$$\alpha = \frac{\|\mathbf{x}_{k+2} - \mathbf{x}_{k+1}\|}{\|\mathbf{x}_{k+1} - \mathbf{x}_k\|},$$

и, если $\alpha < 1$, определяется требуемое число элементов орбиты N , необходимое для определения неподвижной точки \mathbf{x}_0^* с точностью $\varepsilon = s_B$, по формуле

$$N = \left[\ln \frac{s_B (1 - \alpha)}{\|\mathbf{x}_1 - \mathbf{x}_0\|} / \ln \alpha \right] + 1,$$

где через $[x]$ обозначено наибольшее целое, не превосходящее x . Не останавливаясь на выводе этого выражения, отметим, что оно очевидным образом вытекает из соотношений, описывающих скорость сходимости итерационной последовательности, генерируемой сжимающим отображением [7]. После этого определяется требуемое время интегрирования $N \cdot T$ и общее время, равное $N \cdot T \cdot \text{количество оставшихся точек}$, необходимое для завершения задачи. Такой расчёт проводится для каждой трёх последовательных точек, поэтому расчётное время постоянно корректируется и выводится на экран в виде индикатора выполнения задания.

Для графического представления областей притяжения программа выводит все точки зоны сканирования на фазовую плоскость или в трехмерное фазовое пространство с пометкой цветом, какой области притяжения принадлежит каждая точка.

В программе предусмотрена возможность вывода фазовых траекторий или их двух-, трёхмерных проекций, а также, выполнение спектрального анализа стационарных решений. Используется одна из p -норм метрических пространств, а, именно, $\|\mathbf{x}\| = \max_i |x_i|$, область D задается в виде n -мерного параллелепипеда $D = [a_1, b_1; a_2, b_2; \dots a_n, b_n]$. Шаг сканирования при построении ядра притяжения равен s_B , при построении остальной части области притяжения, — s_D . Начальное значение r_0 , — радиуса ядра области притяжения равно s_B . Шаг изменения $\Delta r = r_{k+1} - r_k$ радиуса шара при построении ядра принят равным $\Delta r = s_B$, при построении остальной части области притяжения значение $\Delta r = s_D$.

Вычислительные эксперименты, результаты, анализ

Приведём три примера качественного тестирования программы ВОА. Вычисления проводились на компьютере с процессором Intel Pentium 4 3.2 ГГц и оперативной памятью объёмом 3 ГБ.

Пример 1. Рассматривается уравнение Дуффинга

$$\frac{d^2v}{dt^2} + k \frac{dv}{dt} + v^3 = B \cos t \quad (3)$$

или в нормальной форме система

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -x_1^3 - k x_2 + B \cos t. \end{cases}$$

При значениях параметров $k = 0.2$, $B = 0.3$ оно имеет три периодических решения [2]

$$\begin{aligned} v_1 &= 0.067 \sin t - 0.310 \cos t + 0.001 \sin 3t - 0.001 \cos 3t, \\ v_2 &= 0.671 \sin t - 0.744 \cos t + 0.026 \sin 3t + 0.022 \cos 3t, \\ v_3 &= 0.988 \sin t + 0.684 \cos t + 0.021 \sin 3t - 0.061 \cos 3t, \end{aligned}$$

одно из которых v_2 — неустойчиво. В данном случае период всех решений равен 2π и поэтому значение $T = 2\pi$.

Построим область притяжения устойчивых решений v_1 и v_3 в плоскости $t_0 = 0$. Неподвижные точки T -отображения Пуанкаре, соответствующие этим решениям следующие

$$\mathbf{x}_1^*(0) = (-0.311, 0.070), \quad \mathbf{x}_3^*(0) = (0.623, 1.051).$$

Полученные результаты в области $D = \{(x_1, x_2) : x_1, x_2 \in [-3, 3]\}$ приводятся на рисунке 1а. Шаг сканирования области D выбран равным $s_D = 0.01$, шаг сканирования для построения ядер — $s_B = 0.01$. Общее время, затраченное на выполнение задания в среде MATLAB составило 4574 сек., на языке C++, — 459 сек. В обоих случаях интегрирование системы уравнений проводилось методом Рунге-Кутты 4-ого порядка точности. Временные затраты на построение ядер $B_{x_1}^*$, $B_{x_3}^*$ областей притяжения составили 4% от общих, т.о. основные временные затраты, в данном случае,

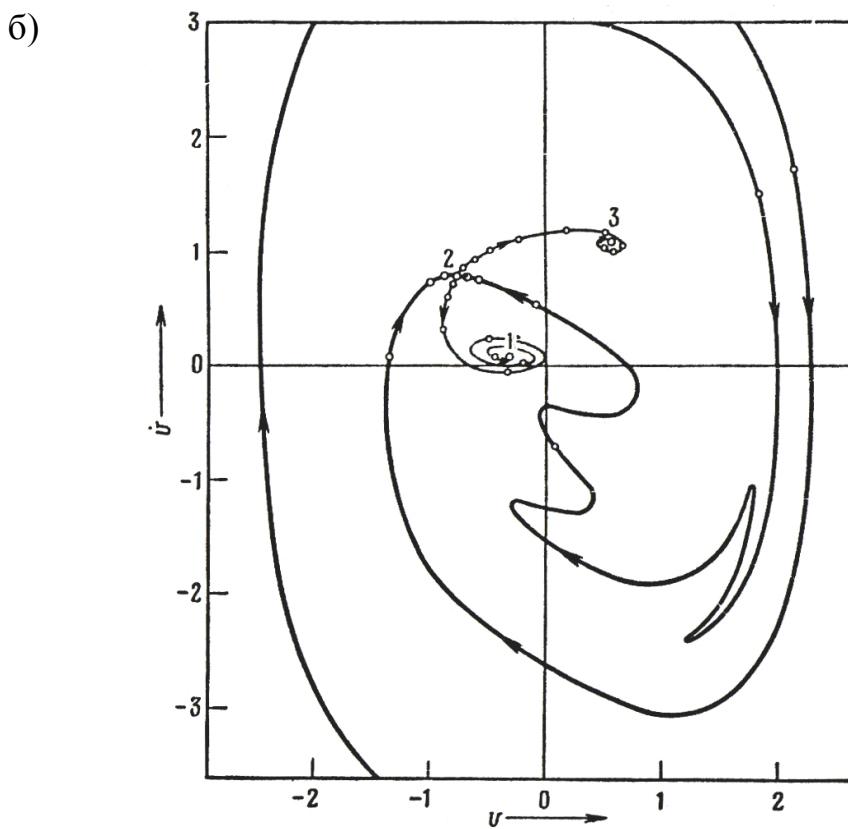
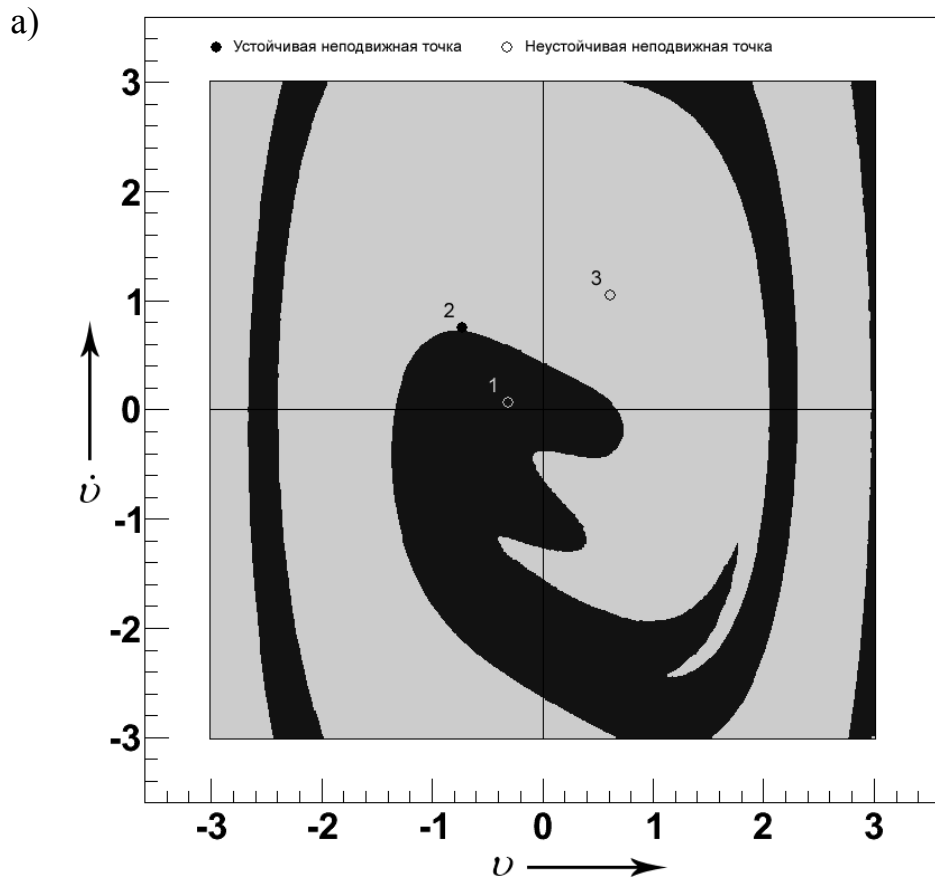


Рисунок 1 — Области притяжения решений уравнения (3) при $k = 0.2$, $B = 0.3$: а) построенные с использованием программы ВОА; б) приведённые в [2]

потребовались на сканирование остальной части. Однако в случае сканирования с шагом $s_D = 0.1$, основные временные затраты, 84% от общих, уже приходится на построение ядер. Это говорит о том, что определение ядер является достаточно весомой частью алгоритма.

На рисунке 1б представлены результаты построения областей притяжения, приведённые в монографии [2]. Построение сепаратрис, разделяющих прямоугольник D на зоны притяжения, выполнялось путём перехода в уравнении (3) к отрицательному времени, последующего его интегрирования по убывающему времени и построения отображения Пуанкаре исходя из неустойчивой неподвижной точки $\mathbf{x}_2^* = (-0.722, 0.748)$, определяемой решением v_2 .

На наш взгляд, приведенные результаты качественно совпадают.

Пример 2. Также рассматривается уравнение (3). При значениях $k = 0.10$, $B = 0.15$ оно уже имеет группу основных режимов, с периодом равным периоду возбуждения

$$\begin{aligned}v_1 &= 0.011 \sin t - 0.153 \cos t, \\v_2 &= 0.806 \sin t - 0.716 \cos t + 0.023 \sin 3t + 0.037 \cos 3t, \\v_3 &= 0.960 \sin t + 0.686 \cos t + 0.019 \sin 3t - 0.040 \cos 3t,\end{aligned}$$

и субгармонических, порядка $1/3$

$$\begin{aligned}v_4 &= 0.063 \sin \frac{1}{3}t + 0.358 \cos \frac{1}{3}t + 0.032 \sin t - 0.180 \cos t, \\v_5 &= -0.342 \sin \frac{1}{3}t - 0.124 \cos \frac{1}{3}t + 0.032 \sin t - 0.180 \cos t, \\v_6 &= 0.278 \sin \frac{1}{3}t - 0.234 \cos \frac{1}{3}t + 0.032 \sin t - 0.180 \cos t, \\v_7 &= 0.149 \sin \frac{1}{3}t + 0.226 \cos \frac{1}{3}t + 0.025 \sin t - 0.171 \cos t, \\v_8 &= -0.271 \sin \frac{1}{3}t + 0.016 \cos \frac{1}{3}t + 0.025 \sin t - 0.171 \cos t, \\v_9 &= 0.122 \sin \frac{1}{3}t - 0.242 \cos \frac{1}{3}t + 0.025 \sin t - 0.171 \cos t,\end{aligned}$$

из которых решения v_2 , v_7 , v_8 , v_9 являются неустойчивыми [2].

Рассматривается $t_0 = 0$, значение $T = 6\pi$, устойчивые неподвижные точки T -отображения Пуанкаре в данном случае следующие

$$\begin{aligned} \mathbf{x}_1^*(0) &= (-0.153, 0.011), \quad \mathbf{x}_3^*(0) = (0.646, 1.016), \\ \mathbf{x}_4^*(0) &= (0.178, 0.053), \quad \mathbf{x}_5^*(0) = (-0.304, -0.082), \quad \mathbf{x}_6^*(0) = (-0.413, 0.124). \end{aligned}$$

Области притяжения, построенные в прямоугольнике $D = \{(x_1, x_2) : x_1, x_2 \in [-3, 3]\}$, представлены на рисунке 2а, значения параметров сканирования $s_D = 0.01$, $s_B = 0.01$. Для сравнения, на рисунке 2б указаны результаты, приведенные в монографии [2], полученные интегрированием уравнения (3) по убывающему времени.

Время выполнения программы в среде MATLAB составило 9574 сек., на языке C++ — 923 сек. В данном случае временные затраты на построение ядер $B_{x_1}^*$, $B_{x_3}^*$, $B_{x_4}^*$, $B_{x_5}^*$, $B_{x_6}^*$ областей притяжения также составили 4% от общих, т.о. основные временные затраты потребовались на сканирование остальной части.

В порядке анализа полученных результатов отметим, что области притяжения имеют вид сужающихся полос, разрывы, заметные на рисунке 2а объясняются тем, что ширина той или иной области притяжения становится меньше шага сканирования s_D . В случае необходимости, уменьшением s_D можно получить уточнённые результаты. Такая возможность продемонстрирована на рисунке 3, где шаг сканирования области D был выбран равным $s_D = 0.005$.

Отметим, что и построение сепаратрис (рисунок 2б) при малой ширине области притяжений также оказывается затруднительным, в силу слишком трудоёмких вычислений. На эти обстоятельства, в частности, указывает и автор [2].

Пример 3. Рассматривается система [3]

$$\begin{cases} \ddot{v} + \delta \dot{v} + v + \beta v(k_{11} v^2 + k_{12} \vartheta^2) = \varepsilon_1 \cos \omega t, \\ \delta \dot{\vartheta} + \gamma \vartheta + \beta \vartheta(k_{21} v^2 + k_{22} \vartheta^2) = \varepsilon_0, \end{cases} \quad (4)$$

которая, очевидно, в нормальной форме представляет систему трёх уравнений первого порядка. Значения параметров следующие

$$\begin{aligned} \delta &= 0.1, \quad \gamma = 1.1, \quad \beta = 0.01, \quad \varepsilon_1 = 2, \quad \varepsilon_0 = 1, \quad \omega = 1.25, \\ k_{11} &= k_{12} = k_{21} = k_{22} = 0.5. \end{aligned}$$

Автор в монографии [3] изучая систему (4), методом медленно меняющихся амплитуд находит периодическое решение с периодом $T = 2\pi/\omega = 1.6\pi$, производит сведение исходной системы к автономному

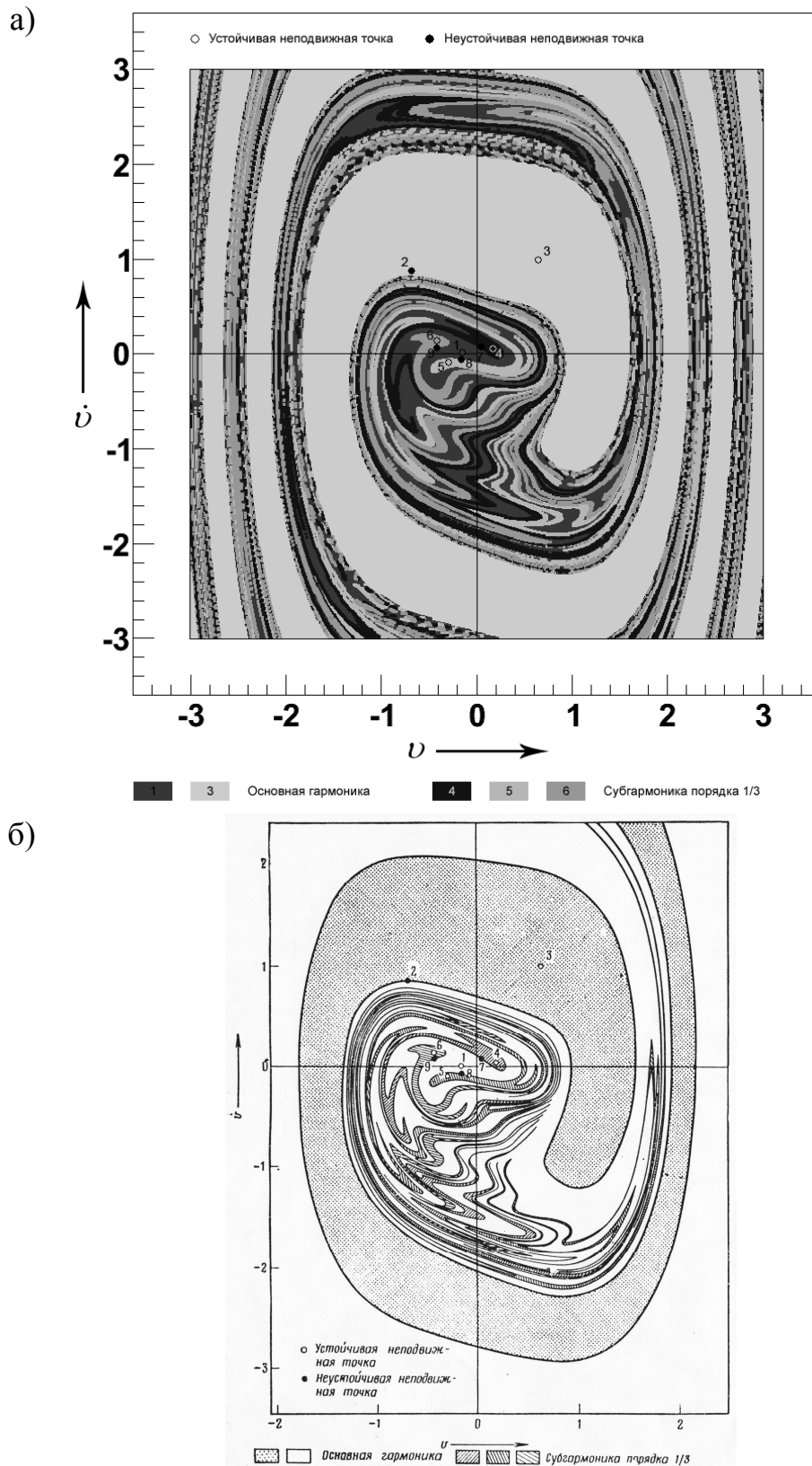


Рисунок 2 — Области притяжения решений уравнения (3) при $k = 0.10$, $B = 0.15$: а) построенные с использованием программы ВОА; б) приведённые в [2]

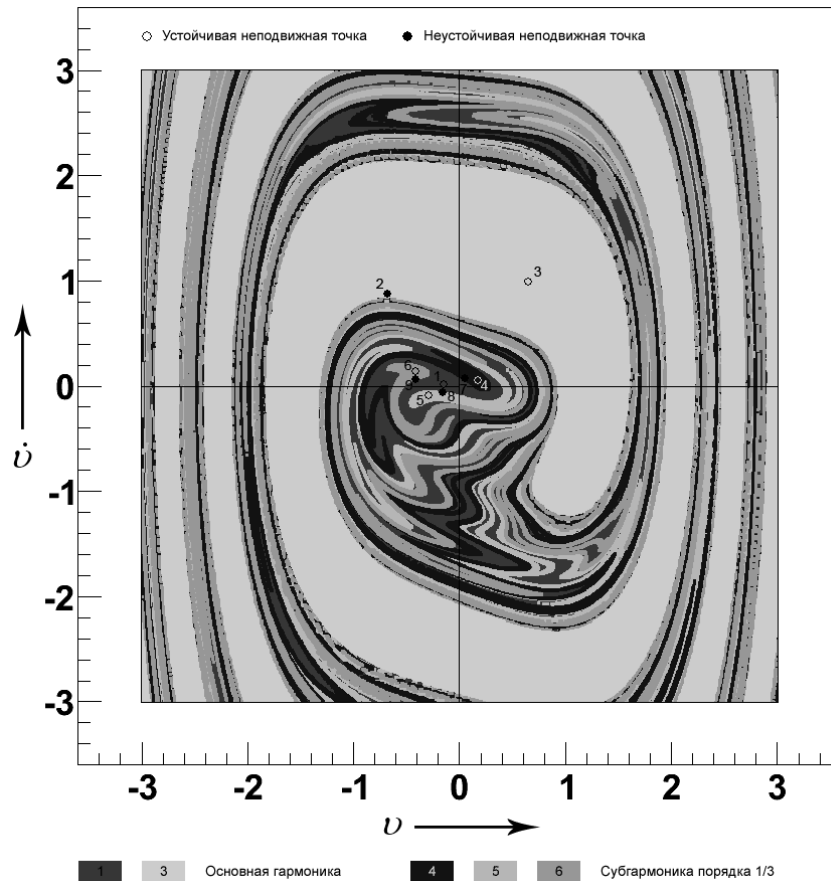


Рисунок 3 — Области притяжения решений уравнения (3), построенные с использованием программы ВОА с уменьшенным шагом сканирования

виду, выполняет классификацию неподвижных точек $\mathbf{x}_1^*(0) = (8.001, 9.778, 0.666)$, $\mathbf{x}_2^*(0) = (-7.931, 6.955, 0.724)$, $\mathbf{x}_3^*(0) = (-3.746, 0.931, 0.876)$ и показывает, что точка \mathbf{x}_2^* является неустойчивой.

На рисунке 4а приводится область притяжения, соответствующая второму режиму, построенная в трёхмерном параллелепипеде $D = \{(x, y, z) : x \in [-26, 22], y \in [-50, 40], z \in [2, 5]\}$, где v , \dot{v} , \mathcal{V} обозначены через x , y , z , соответственно. Шаг сканирования области D был выбран равным $s_D = 0.5$, шаг сканирования для построения ядер — $s_B = 0.25$. На рисунке 4б, для сравнения, показана область притяжения, полученная в [3].

Время выполнения программы в среде MATLAB составило 22519 сек., на языке C++ — 2171 сек. Временные затраты на построение ядер $B_{\mathbf{x}_1^*}$, $B_{\mathbf{x}_3^*}$ областей притяжения составили 41% от общих, т.е. значительные временные затраты, в данном случае, потребовались на сканирование остальной части.

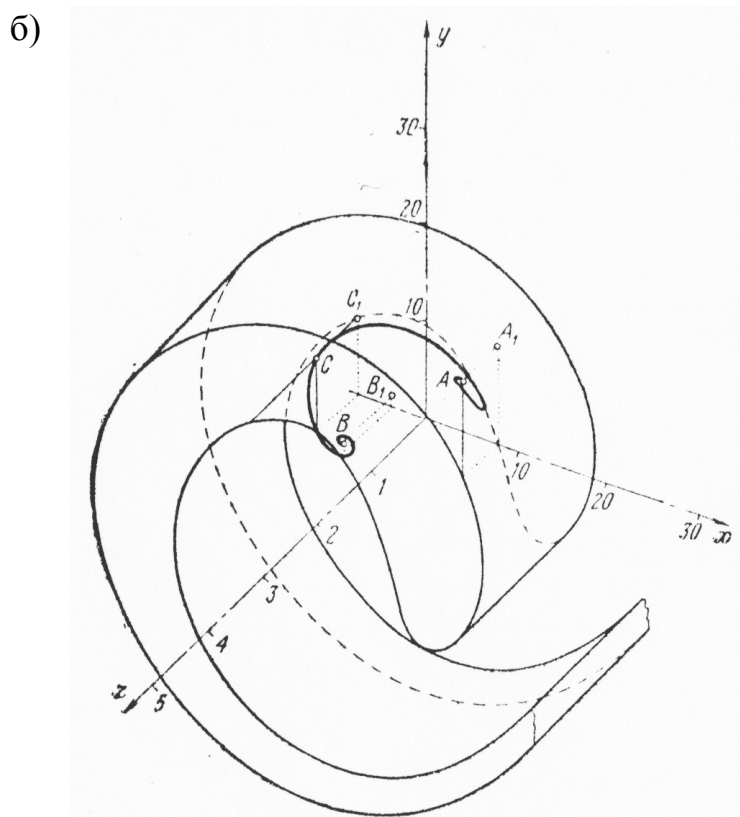
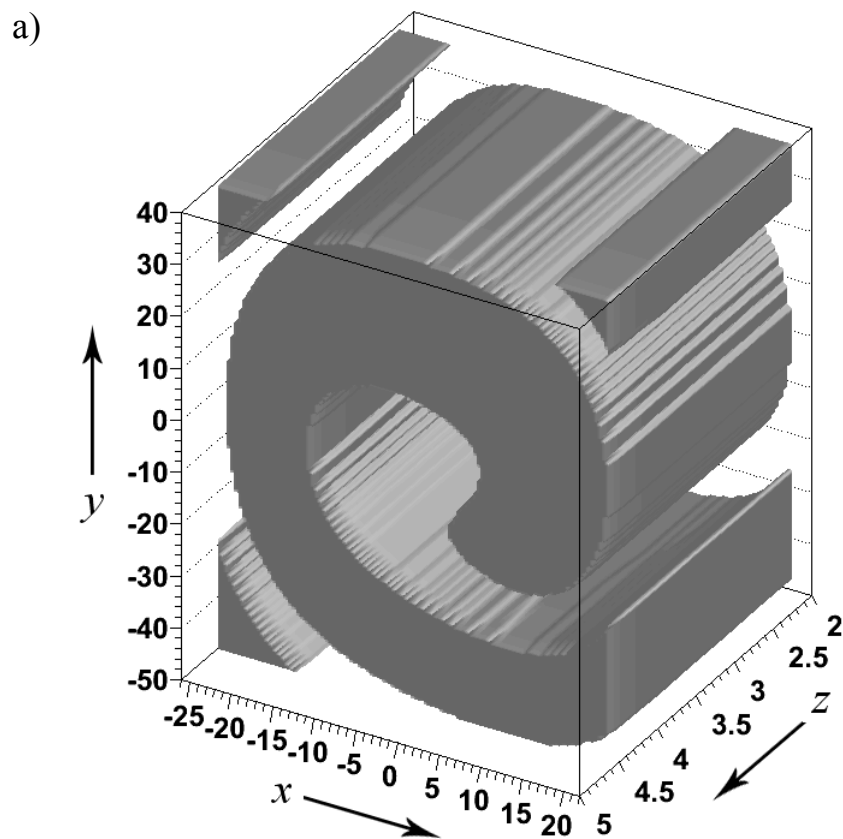


Рисунок 4 — Области притяжения решений системы (4): а) построенные с использованием программы ВОА; б) приведённые в [3]

Заключение

Полученные результаты обнадеживают. Они свидетельствуют о работоспособности описанного алгоритма и демонстрируют высокий уровень качественного соответствия, достигнутый в тестовых примерах. Вместе с тем, значительная ресурсоёмкость, проявленная алгоритмом в сравнительно простых примерах, ставит задачу поиска путей повышения его быстродействия в число очередных. Кроме этого, к числу задач, связанных с дальнейшей отработкой алгоритма, следует отнести также и вопросы определения границ применимости данного метода в части размерности исследуемых систем и анализа систем с негладкими нелинейностями.

Список литературы

1. Thompson J. M. T. Nonlinear Dynamics and Chaos / J. M. T. Thompson, H. B. Stewart. — John Wiley & Sons England, 2002. — 458 p.
2. Хаяси Т. Нелинейные колебания в физических системах: пер. с англ. / Т. Хаяси. — М.: Мир, 1968. — 432 с.
3. Тондл А. Нелинейные колебания механических систем: пер. с англ. / А. Тондл. — М.: Мир, 1973. — 336 с.
4. Зубов В. И. Устойчивость движения (методы Ляпунова и их применение) / В. И. Зубов. — М.: Высшая школа, 1984. — 232 с.
5. Кирин Н. Е. Построение области притяжения по методу Зубова / Н. Е. Кирин, Р. А. Нелепин, В. Н. Байдаев // Дифференциальные уравнения. — 1981. — Т. 17, вып. 8. — С. 1347–1361.
6. Демиденко Г. В. Об устойчивости решений квазилинейных периодических систем дифференциальных уравнений / Г. В. Демиденко, И. И. Матвеева // Сибирский математический журнал. — 2004. — Т. 45, вып. 6. — С. 1271–1284.
7. Кроновер Р. М. Фракталы и хаос в динамических системах. Основы теории: пер. с англ. / Р. М. Кроновер. — М.: Постмаркет, 2000. — 352 с.

Надійшла до редакції 10.01.2010 р. Рецензент: к.ф.-м.н., доц. Клімко Г.Т.

В.М. Бєловодський, М.Ю. Сухоруков

Донецький національний технічний університет

Скануючий алгоритм побудови областей тяжіння періодичних режимів. Розглянутий алгоритм побудови областей тяжіння періодичних режимів, описані особливості його програмної реалізації. Наведені приклади якісного тестування програми, виконано порівняння з відомими результатами.

Область тяжіння, алгоритм побудови, тестування, MATLAB, C++

V.N. Belovodskiy, M.I. Sukhorukov

Donetsk National Technical University

The scanning algorithm for constructing of the domains of attraction of periodic regimes. It is considered the algorithm of constructing of the domains of attraction of periodic regimes and features of its software implementation are described. Examples of software quality testing are given and the received results are compared with known papers.

The domain of attraction, the algorithm for constructing, testing, MATLAB, C++