

§4. Алгоритмы поиска аттракторов

Перейдем теперь к более детальному рассмотрению численных методов поиска аттракторов, и сначала обратимся к еще одному примеру из области вычислительной биологии, а в завершение расскажем о применяемых для этого алгоритмах решения нелинейных алгебраических уравнений.

Пример: логистические модели

Вернемся сначала к самому простому логистическому уравнению (см. 3.7) и выпишем его еще раз:

$$\frac{du}{dt} = u - Bu^2 . \quad (27)$$

Соответственно, алгебраическое уравнение, описывающее положение неподвижных точек, выглядит следующим образом:

$$u - Bu^2 = 0. \quad (28)$$

Уравнение (28) имеет два корня $u_0=0$ и $u_1=1/B$, которые легко отыскать аналитически. Они являются неподвижными точками исходной динамической системы (отметим, типа *устойчивый узел* и *неустойчивый узел* соответственно – а о том, как исследуется устойчивость аттракторов будет рассказано в следующем параграфе).

Сразу подчеркнем: ОДУ (27) зависит от параметра B , что чрезвычайно характерно для задач нелинейной динамики. Поэтому и поиск аттракторов следует осуществлять в зависимости от параметра. С уравнением (28) нам повезло, и мы нашли его точное решение аналитически. Однако, следует иметь в виду, что реальные задачи приходится решать численными методами, и тогда зависимость от параметров (которых, к слову, может быть несколько), существенно усложняет расчеты.

Слегка модифицируем теперь исходное ОДУ (27) с тем, чтобы оно уже не поддавалось такому простому аналитическому

исследованию, и выясним, каким образом можно осуществить те же действия при помощи численных методов.

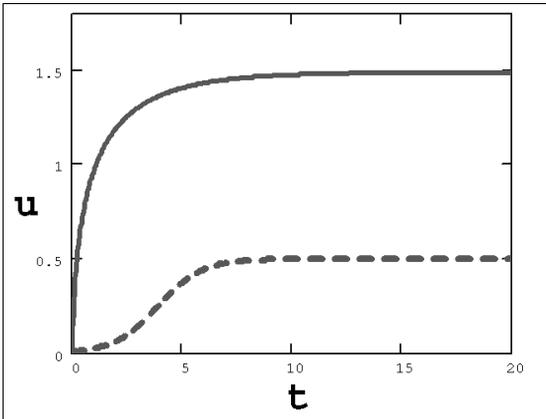


Рис. 16. Решения уравнений (27) (пунктир) и (29) (сплошная кривая)

Рассмотрим следующую модель:

$$\frac{du}{dt} = u - \ln(Bu^2) \quad (29)$$

По-прежнему, B – это параметр модели, причем типичное решение (29) весьма похоже на решение классического логистического уравнения (23) (рис. 16).

Алгебраическое уравнение, задающее положение неподвижных точек, выглядит следующим образом:

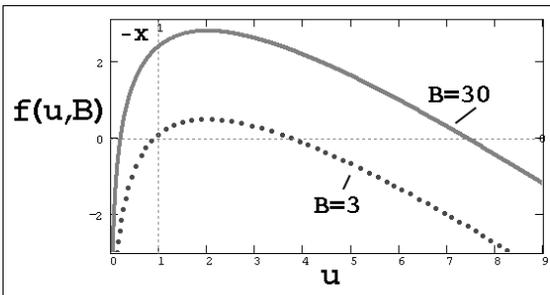


Рис. 17. Геометрический смысл алгебраического уравнения (30)

$$u - \ln(Bu^2) = 0 \quad (30)$$

и уже не поддается точному решению, как это было с уравнением (27). Поэтому решать его, как в большинстве практических случаев, следует численными методами.

Принцип численного решения нелинейных уравнений иллюстрируется рис. 17, на котором представлены графики зависимости $f(u, B) = u - \ln(Bu^2)$ для двух значений параметра B . Две точки пересечения каждого графика с осью X соответствуют выполнению уравнения (30), т. е. их координаты равны искомым корням этого уравнения.

Следует обратить внимание на два важных факта. Во-первых, необходимо получить решение (30) для всевозможных значений параметра модели B , т.е., на самом деле, надо решить не одно алгебраическое уравнение $f(x,B)=0$, а целую серию уравнений. Поскольку они зависят не только от неизвестного u , но и от параметра B , то требуется определить зависимость корня $u^0(B)$.

Во-вторых, количество и, вообще говоря, существование корней для тех или иных B изначально не известно и требует дополнительного исследования. Кстати говоря, сопоставляя кривые на рис. 17, несложно догадаться, что для еще меньших значений B график функции $f(u,B)$ окажется ниже оси X , и это будет означать отсутствие решения для этих B .

Для рассматриваемого класса задач существует множество различных методов, наиболее характерные из которых будут рассмотрены ниже. Мы вначале приведем результат решения задачи (30) в виде зависимости $u^0(B)$ (рис. 18), а затем обсудим детали численных расчетов.

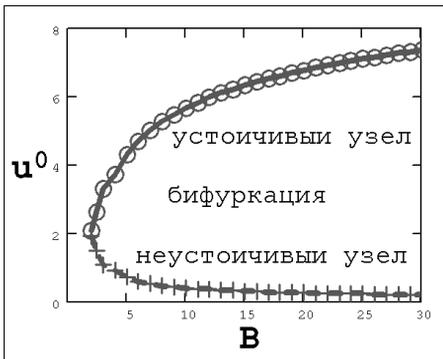


Рис. 18. Решение (30) в зависимости от параметра B

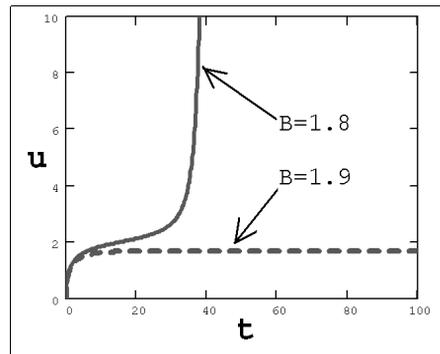


Рис. 19. Решение задач Коши для ОДУ (29) (для двух значений B)

График решения уравнения (30) весьма примечателен и требует дополнительного обсуждения. Прежде всего, следует отметить, что, как и предполагалось, решений для $B < B_0 \approx 1.85$ нет вовсе. В

точке V_0 уравнение (30) имеет единственное решение, а при $V > V_0$ для каждого V имеется пара решений (именно этот случай и иллюстрировали графики на рассмотренном нами выше рис. 17).

Таким образом, значение V_0 является для исходной динамической системы (29) *бифуркационным*, поскольку слева от него неподвижные точки для ОДУ (29) отсутствуют, а справа имеется пара неподвижных точек (тип каждой из которых нам еще предстоит определить). Сказанное иллюстрирует рис. 19, на котором представлено два решения ОДУ (29): для $V < V_0$ и $V > V_0$. В первом случае решение расходится (асимптотически стремясь к бесконечности), а во втором – стремится к аттрактору, имеющемуся у системы при $V > V_0$.

Отметим, что рассмотренный пример был нами исследован исключительно при помощи численных методов, которые позволили изучить фазовый портрет динамической системы в зависимости от параметра, а также определить точку бифуркации.

Остановимся в оставшейся части параграфа на нескольких характерных численных алгоритмах, которые применяются для решения систем нелинейных алгебраических уравнений, возникающих при анализе аттракторов и бифуркаций. Как уже было сказано, поиск стационарного решения системы ОДУ сводится к приравниванию нулю производных в ее левой части, что определяет систему обычных (алгебраических) уравнений. Для системы N ОДУ она была нами выписана в формах (20) и (21).

Отыскание корней системы алгебраических уравнений численными методами связано с двумя задачами. Во-первых, необходимо предварительно локализовать корни, т. е. определить, имеются ли они в принципе, а также исследовать их количество и примерное расположение. Во-вторых, следует провести собственно отыскание корней с заданной погрешностью, т. е. найти значения аргумента, при которых невязки алгебраических уравнений отличаются от нуля не более чем на наперед заданную малую погрешность.

Рассмотрим для простоты случай, когда $N=2$ и явной зависимости от времени нет. Тогда (20) выглядит как:

$$\begin{aligned} f_1(x, y) &= 0, \\ f_2(x, y) &= 0. \end{aligned} \quad (31)$$

Чтобы решить задачу предварительной (грубой) локализации корней, в самых простых случаях можно использовать графическое представление $f_1(x, y)$ и $f_2(x, y)$, что иллюстрируется рис. 20. На нем каждое из двух уравнений (их конкретный вид для данного примера Вы найдете на компакт-диске) показывается в виде зависимости $y(x)$: первое – сплошной кривой, а второе – пунктиром. Точки пересечения кривых соответствуют одновременному выполнению обоих уравнений, т. е. их координаты равны искомым корням системы (31).

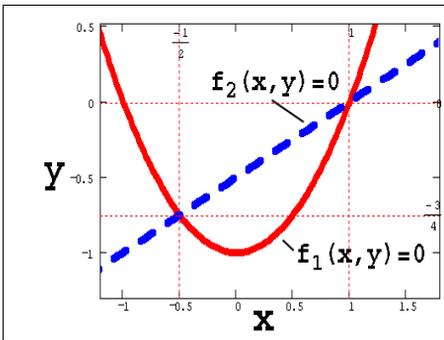


Рис. 20. Графическое решение системы (31)

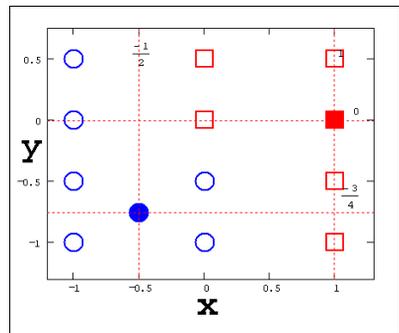


Рис. 21. К объяснению алгоритма сканирования

В случае многомерных систем такой способ практически неприменим. Если требуется исследовать определенную область определения переменных уравнения на наличие корней, определив их примерное положение, то обычно используют весьма расточительный способ, называемый *сканированием*.

Сканирование состоит в последовательном поиске корня, начиная из множества пробных точек, покрывающих расчетную область. На рис. 21 показан фрагмент сетки, узлы которой помечены кружками или квадратами, в зависимости от того, к какому из

двух корней сходится численное решение, если итерации начинаются из данного узла.

Чаще всего сканирование организуют следующим образом. Область определения функции покрывается сеткой (в случае функции двух переменных чаще всего прямоугольной, в случае трех переменных – кубической и т. д.). Из узлов сетки, т.е. каждой элементарной области запускается численный метод поиска корня, и в случае выхода итераций за ее пределы расчеты прерываются, а в противном случае происходит нахождение корня.

Гораздо менее надежной (но зато более экономной) альтернативой является простое вычисление и сравнение между собой невязок системы уравнений в узлах сетки. На тех участках области определения, где норма невязки невелика, вероятность локализации корня больше, и именно из локальных минимумов нормы невязки можно запускать градиентный метод для уточнения корня.

Конечно, гарантии, что все существующие корни будут найдены, особенно в многомерных случаях, чаще всего нет. Всегда существует вероятность «просмотреть» корень, расположенный между узлами сканирования.

Остановимся теперь на конкретных численных алгоритмах, которые применяются для решения алгебраических уравнений. Будем далее говорить об одном уравнении, имея в виду, что всегда возможно обобщение результатов на случай системы уравнений

Один из наиболее простых методов называется *алгоритмом секущих* и состоит (в одномерном случае) в следующем (рис. 22). Сначала выбирается начальное приближение к корню: $x_0 = x$ и рассчитывается следующая точка $x_1 = x_0 + h$, где h – это шаг. Затем вычисляются значения $f_0 = f(x_0)$ и $f_1 = f(x_1)$.

Через эти две точки (x_0, f_0) и (x_1, f_1) проводится секущая – прямая линия, которая пересекает ось X в некоторой точке x_2 . Эта точка

принимается за второе приближение. Новая секущая проводится через точки (x_1, f_1) и (x_2, f_2) , тем самым определяя третье приближение, и т. д.

Если на каком-либо n -м шаге оказывается, что $f(x_n)$ меньше наперед заданной погрешности, то итерационный процесс прерывается, и x_n выдается в качестве решения.

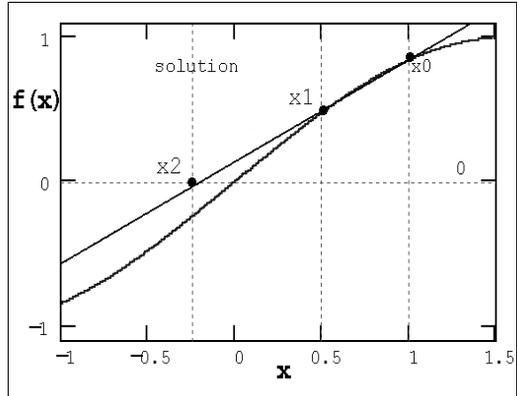


Рис. 22. Алгоритм секущих

Более эффективны градиентные численные алгоритмы, основанных на последовательных приближениях к истинному решению уравнения, которые вычисляются с помощью производной от $f(x)$. Приведем наиболее простую форму алгоритма, называемого *методом Ньютона* (рис. 23).

Во-первых, в точке x_0 , принятой за начальное приближение, методом конечных разностей вычисляется производная $f'(x_0)$. Во-вторых, при помощи разложения Тейлора, строится касательная к графику $f(x)$ в окрестности точки x_0 :

$$y = f(x_0) + f'(x_0) \cdot (x - x_0). \quad (32)$$

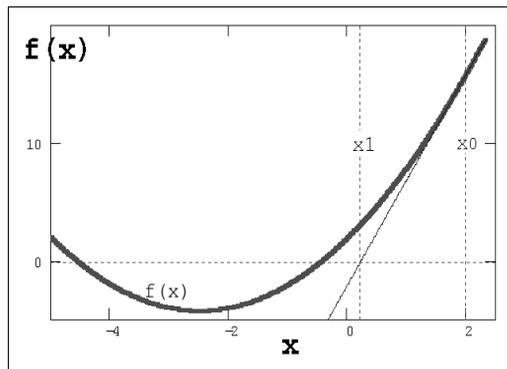


Рис. 23. Алгоритм Ньютона

В-третьих, определяется точка x_1 , в которой прямая пересекает ось X (рис. 23). Если окажется, что $f(x_1)$ меньше установленной погрешности, то итерации прерываются, и значение x_1 выдается в качестве решения. В противном случае x_1 принимается за новую

итерацию, и цикл повторяется: строится касательная к $f(x)$ в точке x_1 , определяется x_2 – точка ее пересечения с осью X и т. д.

В случае системы нескольких уравнений модификация алгоритма Ньютона состоит в линеаризации соответствующих функций многих переменных, т. е. аппроксимации их линейной зависимостью с помощью частных производных.

Решение «хороших» нелинейных уравнений и систем типа тех, которые были рассмотрены ранее, представляет собой несложную, с вычислительной точки зрения, задачу. Как мы уже знаем, для нелинейной динамики характерна более сложная проблема: решение не одного уравнения (или системы), а целой серии уравнений, зависящих от некоторого параметра (или нескольких параметров). Для таких задач существуют очень эффективные методы, которые называются *методами продолжения*.

Суть методов продолжения заключается в следующем. Пусть имеется уравнение

$$f(x, \mu) = 0, \quad (33)$$

зависящее не только от неизвестного x , но и от параметра μ . Требуется определить зависимость корня x от параметра, т. е. $x(\mu)$. Отметим, что с подобным уравнением, а именно (30), мы уже сталкивались, не конкретизируя, каким способом было получено численное решение.

Предположим, что каким-либо методом (например, Ньютона) мы получили корень при $\mu = \mu^*$. Обозначим его как

$$x(\mu^*) = x^*. \quad (34)$$

Конечно, корней может быть несколько, и определить их все можно путем (весьма ресурсоемкой) операции сканирования.

Теперь найти решение для следующего значения $\mu = \mu^* + \Delta\mu$, т. е.

$$f(x, \mu^* + \Delta\mu) = 0, \quad (35)$$

будет гораздо легче, поскольку в качестве хорошего начального приближения (для того же метода Ньютона) можно использовать известное значение x^* . Двигаясь таким образом дальше по параметру μ , мы определим искомую зависимость $x(\mu)$. Если на каждом шаге по μ использовать прием «предиктор-корректор», т.е. экстраполяцию $x(\mu)$, то алгоритм будет работать еще более эффективно.

Представленный метод продолжения по параметру можно оформить более элегантно. Для этого достаточно формально продифференцировать (33) по μ :

$$\frac{df(x, \mu)}{d\mu} = \frac{\partial f(x, \mu)}{\partial x} \cdot \frac{dx}{d\mu} + \frac{\partial f(x, \mu)}{\partial \mu} = 0, \quad (36)$$

откуда следует:

$$\frac{dx}{d\mu} = - \left(\frac{\partial f(x, \mu)}{\partial x} \right)^{-1} \cdot \frac{\partial f(x, \mu)}{\partial \mu} = 0. \quad (37)$$

Соотношение (37) есть ни что иное, как обыкновенное дифференциальное уравнение, решив которое с начальным условием (34), мы получим искомую зависимость $x(\mu)$. Соответствующий пример решения той же задачи (30) в форме задачи Коши Вы найдете на компакт-диске.

Важно заметить, что зависимость $x(\mu)$ может состоять из нескольких ветвей решения, как например, в случае (30) (см. рис. 18). Иногда, если численный метод организован не совсем удачно (например, выбран слишком большой шаг по μ), могут происходить «скачки» с одного семейства решений на другое (рис. 24). По мере увеличения μ кривая корней уравнения $x(\mu)$ сначала плавно идет по одному (нижнему) семейству решений, а потом (в районе $\mu \sim 11$) явно срывается, «перепрыгивая» на другое семейство. С вычислительной точки зрения такая ситуация чаще всего крайне неблагоприятна, поскольку хотелось бы отыскать непрерывное семейство решений. Скачки зависимости $x(\mu)$ могут вводить пользователя в заблуждение, вовсе скрывая от него существование нижнего семейства решений.

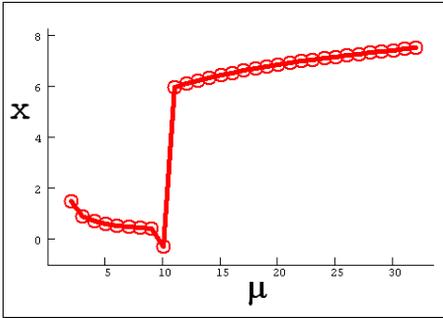


Рис. 24. Срыв численного решения ОДУ (30) с одной ветви на другую

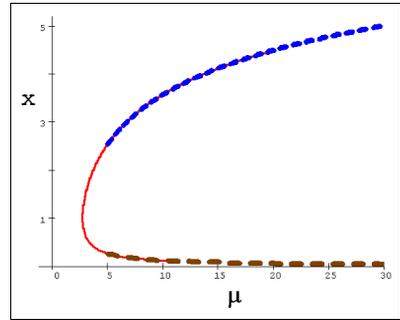


Рис. 25. Три ветви решения (30), найденные согласно (37) и (38)

Скачки наиболее вероятны в области, где ветви решения близки: в нашем случае, в районе точки поворота. Для того, чтобы правильно определить всю кривую $x(\mu)$, в окрестности таких точек достаточно вместо $x(\mu)$ искать зависимость $\mu(x)$, т.е. вместо ОДУ (37) решать (с соответствующим начальным условием, взятым с вычисленного участка кривой) уравнение

$$\frac{d\mu}{dx} = - \left(\frac{\partial f(x, \mu)}{\partial \mu} \right)^{-1} \cdot \frac{\partial f(x, \mu)}{\partial x} = 0 \quad (38)$$

Пример реализации этих соображений показан на рис. 25. На нем Вы видите искомый график зависимости $x(\mu)$, сконструированный из трех ветвей: согласно решению ОДУ (38) (сплошная кривая) и (37) (две пунктирные кривые).

Суммируем результаты параграфа. Сначала серию задач Коши для ОДУ, зависящих от некоторого параметра, мы свели к соответствующей системе алгебраических нелинейных уравнений, опять-таки зависящих от того же параметра; а те, в свою очередь, оформили как новую задачу Коши. С вычислительной точки зрения этот прием представляется весьма выгодным, т.к. решение задач Коши – это довольно простая задача.