Компьютерное моделирование

1 Методы интегрирования уравнения Ньютона

Рассмотрим несколько наиболее известных методов дискретизации дифференциальных уравнений, принадлежащих группе конечно разностных методов.

Будем рассматривать уравнения вида $ma = F$, то есть $\ddot{x} = F(t, x, \dot{x})/m$. Зная $F(t, x, \dot{x})$, можно посчитать ускорение a. Будем решать задачу восстановления траектории и скорости тела при заданной приложенной силе:

$$\begin{cases} \dfrac{dv}{dt} = a, \\ \dfrac{dx}{dt} = v. \end{cases}$$

Для простоты будем расматривать одномерную постановку задачи.

Обычно для создания конечно разностной схемы применяется разложение в ряд Тейлора:

$$x(t + \Delta t) = x(t) + \sum_{i=1}^{n-1} \frac{1}{i!} x^{(i)}(t) \Delta t^i + R_n$$

Целью любого конечно разностных метода является вычисление значений x_{n+1}, v_{n+1} в момент времени $t_{n+1} = t_n + \Delta t$. Шаг Δt выбирается таким образом, чтобы выполнялась асимптотическая сходимость полученного численного решения и его устойчивость.

Термин "устойчивость"нужно понимать следующим образом: малые возмущения начальных значений приводят к небольшим изменениям полученного численно решения, то есть

$$\forall \delta > 0 \; \exists \varepsilon > 0 : |x(t; x_0, v_0) - x(t; x_0 + \delta, v_0 + \delta)| < \varepsilon$$

Один из способов проверки устойчивости метода заключается в контроле величины полной энергии и контроле того, чтобы она не отклонялась от начального значения, в случае, когда полная энергия сохраняется.

Метод Эйлера

Метод Эйлера, которым мы пользовались при рассмотрении всех приведенных выше примеров, аналогичен сохранению в уравнении членов порядка малости $o(\Delta t)$:

$$\begin{cases} \upsilon_{n+1} = \upsilon_n + a_n \Delta t \\ x_{n+1} = x_n + \upsilon_n \Delta t \end{cases}$$

достоинства метода:

1. простота
2. явный метод: для определения нового значения величины используют только ее предыдущие значения

недостатки:

1. метод первого порядка точности;
2. на каждом шаге получена ошибка $O(\Delta t^2)$, за шагов n общая ошибка накапливается: $nO(\Delta t^2)$;
3. схема Эйлера является асимметричной за счет того, при вычислении нового значения координаты используется значение скорости, запаздывающее на один шаг, следовательно, этот метод порождает неустойчивые решения.
4. метод сносовый, т.е. без сохранения полной энергии

Метод Эйлера-Крамера

$$\begin{cases} \upsilon_{n+1} = \upsilon_n + a_n \Delta t \\ x_{n+1} = x_n + \upsilon_{n+1} \Delta t \end{cases}$$

Этот метод также является методом первого порядка точности, но он лучше предыдущего тем, что является симметричным.

Метод средней точки

$$\begin{cases} \upsilon_{n+1} = \upsilon_n + a_n \Delta t, \\ x_{n+1} = x_n + \dfrac{1}{2}(\upsilon_n + \upsilon_{n+1}) \Delta t. \end{cases}$$

$$x_{n+1} = x_n + \frac{1}{2}(\upsilon_n + \upsilon_n + a_n \Delta t)\Delta t = x_n + \frac{1}{2}(2\upsilon_n \Delta t + a_n \Delta t^2) = x_n \upsilon_n \Delta t + \frac{1}{2} a_n \Delta t^2$$

По скорости этот метод имеет первый порядок точности, а по координате – второй. Также сносовый метод

Методы полушага

Здесь средняя скорость на отрезке равна скорости в середине отрезка. Метод полушага относится к методам высокого порядка точности с ограниченной погрешностью.

$$\begin{cases} \upsilon_{n+\frac{1}{2}} = \upsilon_{n-\frac{1}{2}} + a_n \Delta t, \\ x_{n+1} = x_n + \upsilon_{n+\frac{1}{2}} \Delta t \end{cases}$$

Такой метод является несамостартующим, то есть на первом шаге он требует задания дополнительного значения: $\upsilon_{\frac{1}{2}} = \upsilon_0 + \dfrac{1}{2} a_0 \Delta t$. Сносовый метод

Метод Верле

Этот метод "несносовый"(с сохранением энергии), высокого порядка.

$$\begin{cases} x_{n+1} = 2x_n - x_{n-1} + a_n (\Delta t^2), \\ \upsilon_n = \dfrac{x_{n+1} - x_{n-1}}{2\Delta t}. \end{cases}$$

Глобальная погрешность имеет третий порядок точности по координате и второй – по скорости. Метод Верле называют неявной симметрической разностной схемой. Однако, метод несамостартующий, здесь машинная ошибка может перекрыть третий порядок точности.

Метод Верле (скоростной)

$$\begin{cases} x_{n+1} = x_n + \upsilon_n \Delta t + \dfrac{1}{2} a_n \Delta t^2, \\ \upsilon_{n+1} = \upsilon_n + \dfrac{1}{2}(a_{n+1} + a_n) \Delta t. \end{cases}$$

Метод сохраняющий энергию.

Алгоритм Бимана

$$\begin{cases} x_{n+1} = x_n + \upsilon_n \Delta t + \dfrac{1}{6}(4a_n - a_{n-1}) \Delta t^2, \\ \upsilon_{n+1} = \upsilon_n + \dfrac{1}{6}(2a_{n+1} + 5a_n - a_{n-1}) \Delta t. \end{cases}$$

Метод сохраняющий энергию.

Метод предиктора-корректора

Суть метода заключается в следующем. Сначала мы предсказываем (predictor) новое значение координаты:

$$\hat{x}_{n+1} = x_{n-1} + 2\upsilon_n \Delta t.$$

Предсказанное значение координаты позволяет определить ускорение $\hat{a}_{n+1}$. Затем, используя $\hat{a}_{n+1}$, получаем скорректированные (corrector) значения υ_{n+1}, x_{n+1}:

$$\begin{cases} \upsilon_{n+1} = \upsilon_n + \dfrac{1}{2}(\hat{a}_{n+1} + a_n)\Delta t, \\ \\ x_{n+1} = x_n + \dfrac{1}{2}(\hat{\upsilon}_{n+1} + \upsilon_n)\Delta t. \end{cases}$$

Скорректированное значение x_{n+1} используется для вычисления нового предсказанного значения a_{n+1} и, значит, для новых предсказанных значений υ_{n+1}, x_{n+1}. Эта процедура повторяется до тех пор, пока предсказанное и скорректированное значение x_{n+1} не будут отличаються меньше, чем на заданную величину.

Метод Рунге-Кутта 4-го порядка

Метод позволяет решать системы обыкновенных дифференциальных уравнений (ОДУ) первого порядка следующего вида:

$$\begin{aligned} \dot{x} &= f(x, \upsilon, t), \\ \dot{\upsilon} &= g(x, \upsilon, t) \end{aligned}$$

Метод Рунге-Кутта заключается в рекурентном применении следующих формул:

$$x_{n+1} = x_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

$$\upsilon_{n+1} = \upsilon_n + \frac{1}{6}(m_1 + 2m_2 + 2m_3 + m_4),$$

где

$$k_1 = f(x_n, \upsilon_n, t_n)\Delta t,$$

$$m_1 = g(x_n, \upsilon_n, t_n)\Delta t,$$

$$k_2 = f(x_n + \frac{k_1}{2}, \upsilon_n + \frac{m_1}{2}, t_n + \frac{\Delta t}{2})\Delta t,$$

$$m_2 = g(x_n + \frac{k_1}{2}, \upsilon_n + \frac{m_1}{2}, t_n + \frac{\Delta t}{2})\Delta t,$$

$$k_3 = f(x_n + \frac{k_2}{2}, \upsilon_n + \frac{m_2}{2}, t_n + \frac{\Delta t}{2})\Delta t,$$

$$m_3 = g(x_n + \frac{k_2}{2}, \upsilon_n + \frac{m_2}{2}, t_n + \frac{\Delta t}{2})\Delta t,$$

$$k_4 = f(x_n + k_3, \upsilon_n + m_3, t_n + \Delta t)\Delta t,$$

$$m_4 = g(x_n + k_3, \upsilon_n + m_3, t_n + \Delta t)\Delta t.$$