

Математические методы PRADIS

Laduga

июл. 19, 2026

1	Методы PRADIS: руководство по численному моделированию	1
1.1	Введение	1
1.2	Обозначения	1
1.3	1. Постановка задачи	2
1.4	2. Численное интегрирование	5
1.5	3. Решение нелинейного уравнения методом Ньютона	7
1.6	4. Управление шагом интегрирования	11
1.7	5. Автоматическое формирование математической модели	14
1.8	6. Модели элементов	17
1.9	7. Алгоритм выполнения шага интегрирования	20
1.10	8. Разделение функций в вычислительном ядре	25
1.11	ПРИЛОЖЕНИЕ А — формулы методов интегрирования и коэффициенты приведения якобиана	25
2	Управление точностью интегрирования	29
2.1	Управление точностью интегрирования по времени (DRLTX, DABSX)	29
2.2	Управление шагом интегрирования (SMAX, SMIN, CONTROL, WEIGHT)	31
2.3	Замечания по взаимодействию CONTROL, WEIGHT и точности	33
2.4	Управление точностью решения системы нелинейных уравнений (DRLTU, DABSU, DRLTI, DABSI)	34
2.5	Взаимодействие параметров точности	37

Методы PRADIS: руководство по численному моделированию

1.1 Введение

Данный документ предназначен для инженеров-пользователей **PRADIS** и разработчиков моделей элементов, которым необходимо понимать общую логику работы вычислительного ядра **PRADIS** без углубленного изучения теории численных методов.

Основная задача документа - показать:

- как математическая модель преобразуется в расчетную схему;
- как выполняется шаг интегрирования;
- как формируется система нелинейных алгебраических уравнений;
- как применяется метод Ньютона;
- какую роль играют модели элементов;
- как обеспечивается универсальность вычислительного ядра **PRADIS**.

Весь материал изложен на сквозном числовом примере с конкретными итерациями, коэффициентами и критериями останова.

1.2 Обозначения

Таблица 1: Основные обозначения, используемые в документе

Символ	Единица измерения (в при- мере)	Описание
x	м	Перемещение (координата) тела
ν	м/с	Скорость тела
a	м/с ²	Ускорение тела
z	м/с	Базисная переменная (в методе Ньютона — скорость ν_i)
F^c	Н	Внешняя синусоидальная сила
F^y	Н	Сила упругости пружины
F^b	Н	Сила вязкого сопротивления
F^i	Н	Сила инерции ($-ma$)
δ_f	Н	Критерий останова по невязке уравнения
δ_z	м/с	Критерий останова по приращению переменной
δ_i	м/с	Допустимая локальная погрешность
lp_i	м/с	Оценка локальной погрешности на шаге i

1.3 1. Постановка задачи

На примере колебательной системы с одной степенью свободы показана **ключевая идея алгоритма вычисленного ядра PRADIS**: как исходная элементная схема (пружина, демпфер, масса, внешняя сила) автоматически преобразуется на каждом шаге по времени в систему нелинейных алгебраических уравнений относительно скорости, и как эта система решается методом Ньютона с автоматическим управлением шагом.

Весь материал изложен на **сквозном числовом примере**, который последовательно проводится через три шага интегрирования:

- **Шаг 1** (раздел 3.3) — первый шаг интегрирования от t_0 до t_1 . Демонстрируется решение нелинейного уравнения скорости методом Ньютона без управления шагом.
- **Шаг 2** (раздел 4.3) — второй шаг интегрирования от t_1 до t_2 . Показывается оценка локальной погрешности и автоматическое уменьшение шага при превышении допустимой погрешности.
- **Шаг 3** (раздел 7) — третий шаг интегрирования от t_2 до t_3 . Иллюстрируется повторное применение управления шагом в более сложной ситуации.

Результаты каждого шага служат начальными условиями для следующего, что позволяет проследить поведение алгоритма в динамике. Итоговая сводка результатов приведена в конце раздела 7.

1.3.1 1.1 Описание физической системы

Рассматривается колебательная система с одной степенью свободы: тело массы m , соединённое с неподвижным основанием упругой пружиной. Движение происходит в среде с вязким сопротивлением под действием внешней силы (рис. 1.1). Начальные условия: заданы положение тела и его начальная скорость.

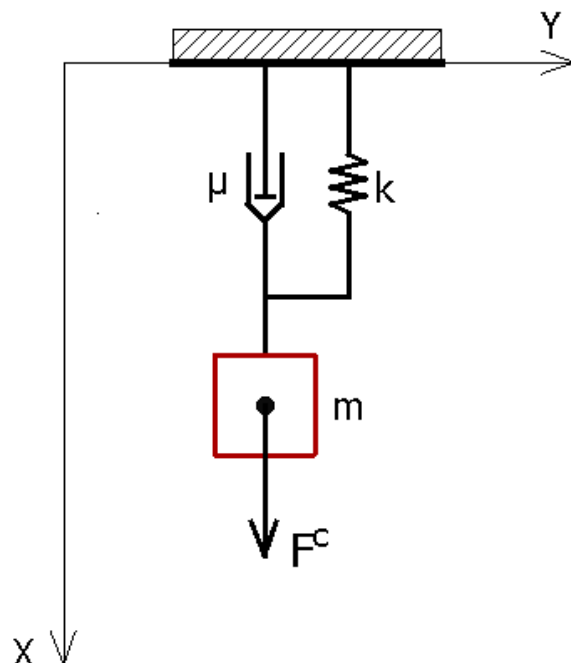


Рис. 1: Рис. 1.1 — Колебательная система

Требуется определить перемещение $x(t)$, скорость $v(t)$ и ускорение $a(t)$ на интервале $[t_0, t_{\text{конеч}}]$.

1.3.2 1.2 Характеристики модели

В качестве иллюстрации решения нелинейной задачи примем следующие модели элементов: демпфер — с квадратичным трением (рис. 1.2.б), пружина — линейная (рис. 1.2.а), внешняя нагрузка — гармоническая (рис. 1.2.в). Их характеристики приведены на рис.1.2

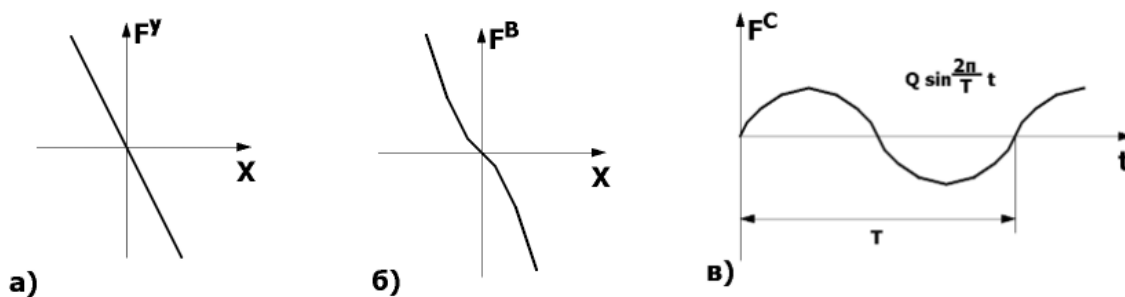


Рис. 2: Рис. 1.2 — Характеристики элементов: а) упругая сила, б) сила вязкого сопротивления, в) внешняя сила

Каждый из этих элементов создаёт свою силу:

- Сила упругости пружины (рис. 1.2, а) — линейна: $F^y = -kx$
- Сила вязкого сопротивления (рис. 1.2, б) пропорциональна квадрату относительной скорости: $F^B = -\mu\nu|\nu|$
- Внешняя сила (рис. 1.2, в) — синусоидальная: $F^c = Q \sin(2\pi t/T)$

В соответствии с принципом Даламбера, добавим силу инерции:

$$F^и = -ma$$

Таким образом, все силы, действующие на систему, можно записать в виде системы уравнений:

$$\begin{cases} F^c = Q \sin \frac{2\pi}{T} t \\ F^y = -kx \\ F^B = -\mu\nu |\nu| \\ F^и = -ma \end{cases} \quad (1.1)$$

1.3.3 Дифференциальное уравнение движения

Зависимости, позволяющие получить дифференциальное уравнение движения, в соответствии со вторым законом Ньютона, имеют вид:

$$ma = F^c + F^y + F^B \quad (1.2)$$

Удобной для численного решения является форма равновесия, где сумма всех сил (включая силу инерции) равна нулю:

$$F^c + F^y + F^B + F^и = 0, \quad F^и = -ma \quad (1.3)$$

На рис. 1.3 показаны положительные направления действующих сил.

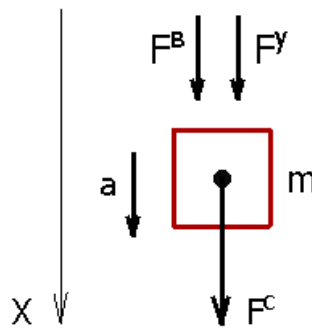


Рис. 3: Рис. 1.3 — Направление сил

Подстановка соотношений для сил (раздел 1.2) в формулу равновесия (1.3):

$$kx + \mu\nu|\nu| + ma - Q \sin \frac{2\pi}{T} t = 0 \quad (1.4)$$

Учитывая, что

$$\nu = \frac{dx}{dt}, \quad a = \frac{d^2x}{dt^2}, \quad (1.5)$$

получаем дифференциальное уравнение движения тела:

$$kx + \mu \frac{dx}{dt} \left| \frac{dx}{dt} \right| + m \frac{d^2x}{dt^2} - Q \sin \frac{2\pi}{T} t = 0 \quad (1.6)$$

1.4 2. Численное интегрирование

1.4.1 2.1 Дискретизация времени

Использование численного подхода к интегрированию уравнения (1.5) предполагает нахождение приближенного решения для определенных моментов времени. Временная ось представляется совокупностью точек $t_0, t_1, t_2, \dots, t_i, t_{i+1}, \dots, t_n$ (рис. 2.1), в каждой из которых ищется приближенное решение.

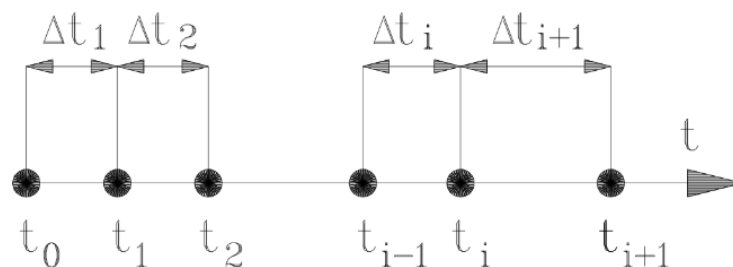


Рис. 4: Рис. 2.1 — Дискретизация временной оси

Интегрирование осуществляется последовательно; выбор величины очередного шага Δt_i зависит как от требуемых показателей точности, так и от результатов интегрирования на предыдущих шагах.

Таким образом, использование численного подхода позволяет перейти от непрерывных значений x, ν, a на всем промежутке времени от t_0 до $t_{\text{конеч}}$ к совокупности дискретных значений x_i, ν_i, a_i для определенных моментов времени t_i . При этом алгебраические формулы выбранного метода интегрирования заменяют дифференциальные соотношения (1.4).

1.4.2 2.2 Неявная одношаговая схема интегрирования уравнений движения

В вычислительном ядре **PRADIS** для интегрирования уравнений движения используется семейство неявных одношаговых методов. Общая идея таких методов заключается в том, что значения переменных в новый момент времени t_i вычисляются с использованием информации о состоянии системы в тот же момент t_i , что и делает схему **неявной**.

В отличие от явных методов, где новое состояние вычисляется непосредственно по известным значениям с предыдущего шага, в неявных методах интересующие нас величины x_i, ν_i, a_i оказываются связанными между собой и с ранее известными значениями $x_{i-1}, \nu_{i-1}, a_{i-1}$ через систему алгебраических соотношений. Эти соотношения зависят от выбранного метода интегрирования и величины шага Δt_i .

В качестве основной схемы в настоящем руководстве рассматривается **метод Штермера**. Этот метод является одним из наиболее распространённых в задачах структурной динамики благодаря своей устойчивости и хорошим энергетическим свойствам. Для системы с одной степенью свободы метод Штермера задаёт связь между переменными на новом и предыдущем шагах следующим образом:

$$\begin{cases} x_i = x_{i-1} + \nu_{i-1}\Delta t_i + a_i \frac{\Delta t_i^2}{2} \\ \nu_i = \nu_{i-1} + a_i \Delta t_i \end{cases} \quad (2.1)$$

Метод является **неявным**, так как ускорение a_i в момент t_i неизвестно заранее и входит в правую часть обоих уравнений. Подстановка (2.1) в уравнение движения приводит к нелинейному алгебраическому уравнению относительно a_i (или, что эквивалентно, относительно ν_i), которое решается итерационно.

В состав вычислительного ядра **PRADIS** также включены и другие неявные разностные схемы, такие как неявный метод Эйлера, метод трапеций и метод Ньюмарка. Их применение может быть предпочтительным в зависимости от специфики решаемой задачи (например, для систем с сильным демпфированием или для задач, где важна сохранность энергии). Справочные формулы для этих методов, а также соответствующие коэффициенты приведения производных, приведены в **Приложении А**.

1.4.3 2.3 Реализация нелинейных систем

Для любого из перечисленных методов интегрирования последовательность действий при численном решении остаётся единой:

1. Уравнение равновесия в момент t_i дополняется выбранными формулами интегрирования, связывающими x_i , ν_i и a_i с известными значениями с предыдущего шага.
2. С использованием этих формул все неизвестные переменные выражаются через одну базисную переменную (как правило, это скорость ν_i или ускорение a_i). В результате формируется одно нелинейное алгебраическое уравнение относительно выбранной базисной переменной.
3. Полученное уравнение решается итерационно (например, методом Ньютона-Рафсона) до достижения сходимости с заданной точностью. На каждой итерации вычисляются значение функции (невязка уравнения) и её производная по базисной переменной.
4. После нахождения базисной переменной по формулам интегрирования восстанавливаются остальные кинематические характеристики (x_i и a_i), после чего выполняется оценка локальной погрешности и, при необходимости, корректировка шага интегрирования.

В дальнейшем при подробном численном разборе в качестве примера выбран **метод Штермера**. Для остальных методов, перечисленных в Приложении А, порядок действий и логика работы вычислительного ядра полностью аналогичны; отличие заключается лишь в конкретном виде формул связи (2.1) и, как следствие, в коэффициентах приведения производных.

1.4.4 2.4 Система уравнений для одного шага

Определим значения x_1 , ν_1 , a_1 для момента времени $t_1 = t_0 + \Delta t_1$.

Уравнение (1.4) для момента времени t_i имеет вид:

$$kx_i + \mu\nu_i |\nu_i| + ma_i - Q \sin \frac{2\pi}{T} t_i = 0 \quad (2.2)$$

Дополняя это соотношение формулами метода интегрирования (2.1), получаем для момента времени t_1 замкнутую систему уравнений:

$$\begin{cases} kx_1 + \mu\nu_1 |\nu_1| + ma_1 - Q \sin \frac{2\pi}{T} t_1 = 0 \\ x_1 = x_0 + \nu_0 \Delta t_1 + a_1 \frac{\Delta t_1^2}{2} \\ \nu_1 = \nu_0 + a_1 \Delta t_1 \end{cases} \quad (2.3)$$

1.4.5 2.5 Сведение к одному уравнению относительно скорости

Приведем полученную систему (2.3) к одному уравнению, выразив неизвестные x_1 и a_1 через ν_1 с помощью формул метода Штермера:

$$\begin{cases} a_1 = \frac{\nu_1 - \nu_0}{\Delta t_1} \\ x_1 = x_0 + \frac{\nu_0 + \nu_1}{2} \Delta t_1 \end{cases} \quad (2.4)$$

Подставляя выражения для a_1 и x_1 из (2.1) в исходное уравнение движения (1.4), получаем:

$$k \left(x_0 + \frac{\nu_0 + \nu_1}{2} \Delta t_1 \right) + \mu\nu_1 |\nu_1| + m \frac{\nu_1 - \nu_0}{\Delta t_1} - Q \sin \frac{2\pi}{T} t_1 = 0$$

Сгруппируем слагаемые относительно ν_1 и $|\nu_1|$:

$$\mu\nu_1 |\nu_1| + \left(\frac{k\Delta t_1}{2} + \frac{m}{\Delta t_1} \right) \nu_1 + \left(kx_0 + \frac{k\nu_0 \Delta t_1}{2} - \frac{m\nu_0}{\Delta t_1} - Q \sin \frac{2\pi}{T} t_1 \right) = 0$$

Введем обозначения:

$$\alpha = \mu, \quad \beta = \frac{k\Delta t_1}{2} + \frac{m}{\Delta t_1}, \quad \gamma = kx_0 + \frac{k\nu_0 \Delta t_1}{2} - \frac{m\nu_0}{\Delta t_1} - Q \sin \frac{2\pi}{T} t_1 \quad (2.5)$$

В результате дискретизации получено нелинейное алгебраическое уравнение относительно неизвестной скорости ν_1 на первом шаге интегрирования, которое назовём **уравнением скорости**:

$$\alpha\nu_1 |\nu_1| + \beta\nu_1 + \gamma = 0 \quad (2.6)$$

Уравнение скорости (2.6) сохраняет свою форму для любого момента времени t_i при замене индексов ($1 \rightarrow i$, $0 \rightarrow i-1$), при этом коэффициенты alpha, beta, gamma зависят от шага и известных с предыдущего шага величин.

Таким образом, применение описанного метода интегрирования сводит исходное дифференциальное уравнение (2.2) к последовательности уравнений скорости вида (2.6), решаемых на каждом шаге по времени.

1.5 3. Решение нелинейного уравнения методом Ньютона

Перейдем к решению уравнения (2.6). Для этого используем метод Ньютона.

1.5.1 3.1 Постановка задачи и геометрическая интерпретация

Рассматривается уравнение вида:

$$f(z) = 0 \quad (3.1)$$

где $f(z)$ — нелинейная функция относительно неизвестного z . Корнем уравнения (3.1) называется такое значение $z = z^*$, при котором функция $f(z)$ обращается в ноль.

Геометрически корень z^* соответствует точке пересечения графика функции $f(z)$ с горизонтальной осью z . Таким образом, решение уравнения (3.1) сводится к отысканию абсциссы точки пересечения кривой $f(z)$ с осью z .

Для численного нахождения этого корня применяется метод Ньютона. На каждой j -й итерации метода решение заменяется отысканием точки пересечения касательной к кривой $f(z)$ с осью z ; касательная строится для $z = z^{j-1}$.

На рисунке 3.1 приведена геометрическая интерпретация описанного выше итерационного процесса.

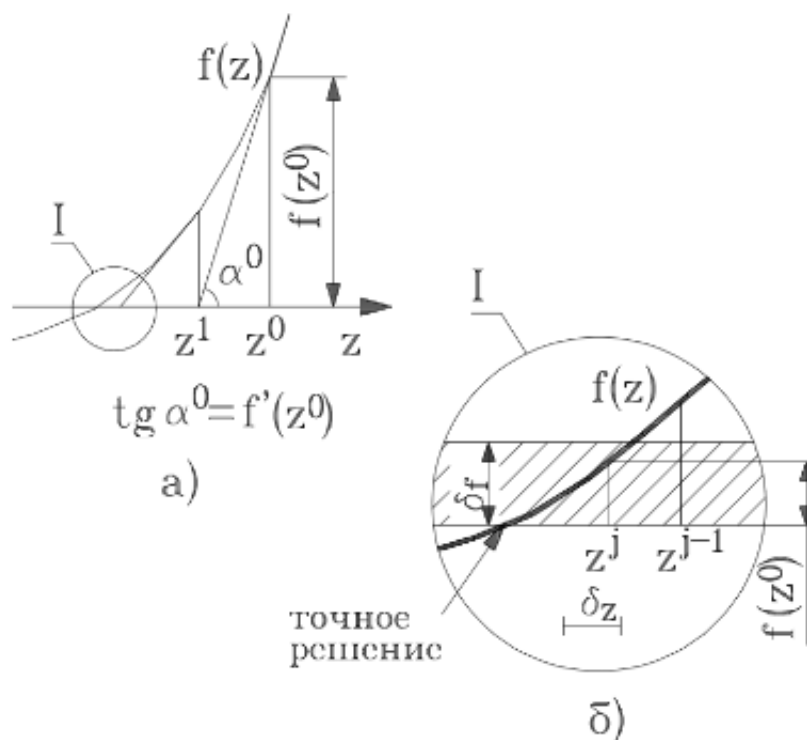


Рис. 5: Рис. 3.1 — Геометрическая интерпретация метода Ньютона

1.5.2 3.2 Алгоритм метода Ньютона

Алгоритм численного решения включает следующие шаги:

1. Выбор начального приближения к решению — величины z^0 .
2. Организация последовательности итераций, на каждой из которых уточняется по-

лученное на предыдущей итерации значение z по схеме (рис. 3.1, а):

$$\begin{cases} z^j = z^{j-1} + \Delta z^j \\ \Delta z^j = -\frac{f(z^{j-1})}{f'(z^{j-1})} \end{cases} \quad (3.2)$$

где $f(z^{j-1})$ — значение функции $f(z)$ при $z = z^{j-1}$, $f'(z^{j-1})$ — значение производной df/dz при $z = z^{j-1}$.

3. Проверка на каждой итерации условия прекращения итераций (рис. 3.1, б):

$$|\Delta z^j| \leq \delta_z, \quad |f(z^j)| \leq \delta_f \quad (3.3)$$

где δ_f — допустимая невязка (отклонение от нуля) правой части уравнения (3.1); δ_z — допустимая величина отличия решения на двух соседних итерациях.

4. Проверка ограничения на максимально допустимое количество итераций:

$$j \leq j_{\max} \quad (3.4)$$

1.5.3 3.3 Пример численного расчёта (Шаг 1)

Выполним **первый шаг интегрирования** — от t_0 до t_1 . Все обозначения соответствуют $i = 1$.

Применим описанный выше метод Ньютона для решения уравнения (2.6). Для этого перепишем его в виде (3.1), приняв $z = \nu_1$:

$$f(z) = \alpha z|z| + \beta z + \gamma \quad (3.5)$$

Производная функции (3.5):

$$f'(z) = 2\alpha|z| + \beta \quad (3.6)$$

Коэффициенты α, β, γ определяются через параметры механической системы согласно (2.1) и (2.6).

Исходные данные для шага 1:

$$k = 20000 \text{ Н/м}, \quad \mu = 1000 \text{ Н} \cdot \text{с}^2 / \text{м}^2, \quad m = 0.1 \text{ кг}, \quad Q = 1000, \quad T = 0.2\pi$$

$$x_0 = 0, \quad \nu_0 = 0, \quad \Delta t_1 = 0.001 \text{ с}$$

Тогда, согласно (2.1):

$$\alpha = \mu = 1000$$

$$\beta = \frac{k\Delta t_1}{2} + \frac{m}{\Delta t_1} = \frac{20000 \cdot 0.001}{2} + \frac{0.1}{0.001} = 10 + 100 = 110$$

$$\gamma = kx_0 + \frac{k\nu_0\Delta t_1}{2} - \frac{m\nu_0}{\Delta t_1} - Q \sin \frac{2\pi}{T} t_1$$

Подставляя значения:

$$t_1 = \Delta t_1 = 0.001 \text{ с}, \quad \frac{2\pi}{T} = \frac{2\pi}{0.2\pi} = 10, \quad \frac{2\pi}{T} t_1 = 10 \cdot 0.001 = 0.01 \text{ рад}$$

Таким образом:

$$\gamma = 20000 \cdot 0 + \frac{20000 \cdot 0 \cdot 0.001}{2} - \frac{0.1 \cdot 0}{0.001} - 1000 \sin(0.01) \approx -10$$

Итак, коэффициенты уравнения скорости:

$$\alpha = 1000, \quad \beta = 110, \quad \gamma = -10$$

Следовательно:

$$f(z) = 1000 z |z| + 110 z - 10, \quad f'(z) = 2000 |z| + 110$$

Зададим параметры метода Ньютона:

$$\delta_z = 0.001, \quad \delta_f = 0.1, \quad j_{\max} = 5, \quad z^0 = 0$$

Первая итерация.

$$\Delta z^1 = -\frac{f(z^0)}{f'(z^0)} = -\frac{1000 \cdot 0 \cdot 0 + 110 \cdot 0 - 10}{2000 \cdot |0| + 110} = \frac{10}{110} = 0.09091$$

$$z^1 = 0 + 0.09091 = 0.09091$$

Проверка критериев остановки:

$$|\Delta z^1| = 0.09091 > \delta_z, \quad f(z^1) = 1000 \cdot 0.09091 \cdot 0.09091 + 110 \cdot 0.09091 - 10 = 8.26 > \delta_f$$

Продолжаем итерации.

Вторая итерация.

$$\Delta z^2 = -\frac{f(z^1)}{f'(z^1)} = -\frac{8.26}{2000 \cdot 0.09091 + 110} = -0.02831$$

$$z^2 = 0.09091 - 0.02831 = 0.06260$$

Проверка: $|\Delta z^2| = 0.02831 > \delta_z$, $|f(z^2)| = 0.80 > \delta_f$.

Третья итерация.

$$\Delta z^3 = -0.00341, \quad z^3 = 0.05918$$

$$|\Delta z^3| = 0.00341 > \delta_z, \quad |f(z^3)| = 0.012 < \delta_f$$

Четвёртая итерация.

$$\Delta z^4 = -0.00005, \quad z^4 = 0.05913$$

$$|\Delta z^4| = 0.00005 < \delta_z, \quad |f(z^4)| = 0.0006 < \delta_f$$

Оба условия (3.3) удовлетворены, ограничение (3.4) не превышено. Решение достигнуто. Скорость для момента времени t_1 :

$$\nu_1 = 0.05913 \text{ м/с}$$

Используя формулы (2.1), определим ускорение и перемещение:

$$a_1 = \frac{\nu_1 - \nu_0}{\Delta t_1} = \frac{0.05913 - 0}{0.001} = 59.13 \text{ м/с}^2$$

$$x_1 = x_0 + \frac{\nu_0 + \nu_1}{2} \Delta t_1 = 0 + \frac{0 + 0.05913}{2} \cdot 0.001 = 2.96 \times 10^{-5} \text{ м}$$

Результат шага 1:

$$x_1 = 2.96 \times 10^{-5} \text{ м}, \quad \nu_1 = 0.05913 \text{ м/с}, \quad a_1 = 59.13 \text{ м/с}^2$$

Примечание

Полученные значения x_1 , ν_1 , a_1 будут использованы как начальные условия для **шага 2** (раздел 4.3).

1.6 4. Управление шагом интегрирования

При использовании описанного выше метода интегрирования (раздел 2) возникает необходимость автоматического выбора шага по времени для обеспечения заданной точности. Для этого используется оценка локальной погрешности.

1.6.1 4.1 Локальная погрешность

Погрешность, возникающую при интегрировании на текущем i -м шаге (её называют локальной), мы будем оценивать по формуле:

$$lp_i = \left| \frac{\nu_i^p - \nu_i^c}{2} \right| \quad (4.1)$$

Расшифруем новые обозначения:

1. ν_i^p — это прогноз (предсказание) скорости, полученный по явной формуле, знакомой нам по выбору начального приближения (4.1):

$$\nu_i^p = \nu_{i-1} + a_{i-1} \Delta t_i \quad (4.2)$$

2. ν_i^c — это скорректированное значение скорости, которое находится из неявной формулы с помощью итераций (например, методом Ньютона):

$$\nu_i^c = \nu_{i-1} + a_i \Delta t_i \quad (4.3)$$

Обратимся к рис. 3.2, который поясняет эту оценку.

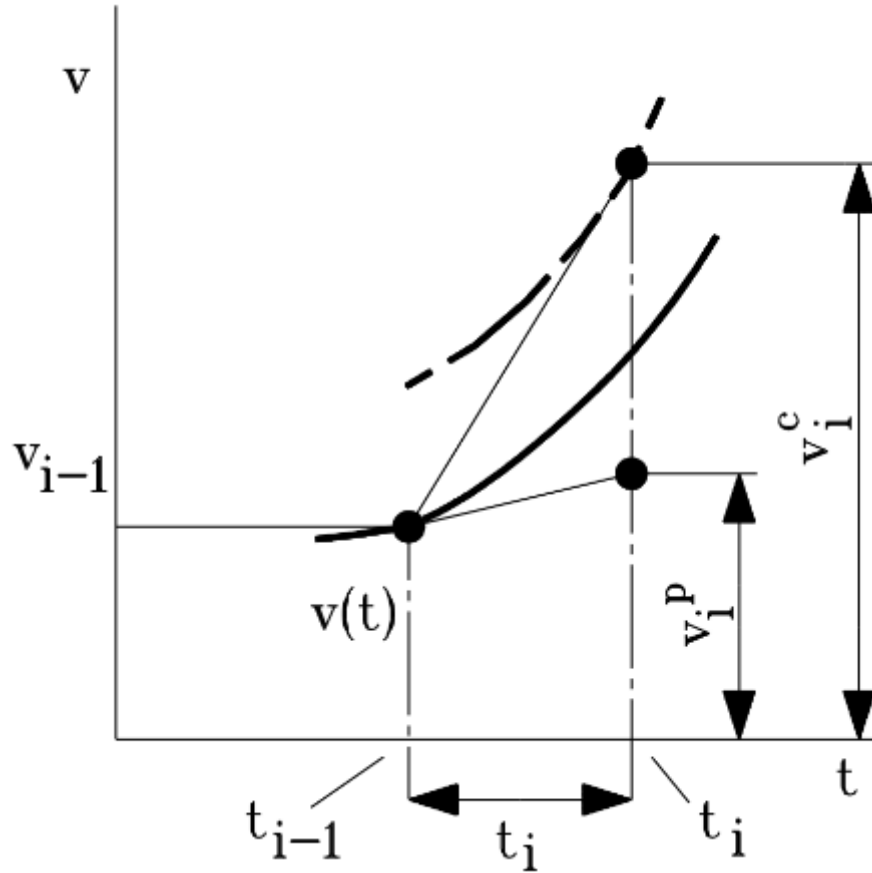


Рис. 6: Рис. 4.4

Допустим, в момент t_{i-1} мы знаем точное значение v_{i-1} . Если мы воспользуемся явной формулой (4.2), мы просто продолжим движение по касательной, проведенной к графику $v(t)$ в точке t_{i-1} . Угол наклона этой касательной задаётся величиной a_{i-1} .

Для расчёта по неявной формуле (4.3) нам понадобится значение a_i — то есть наклон касательной уже в конечной точке шага t_i , который нам пока неизвестен. Поэтому мы решаем систему уравнений (2.1) и (4.3) итеративно.

Как видно из рисунка, истинная кривая $v(t)$ проходит между двумя полученными точками: явным прогнозом v_i^p и уточнённым решением v_i^c . Разность между ними показывает, насколько сильно график скорости изогнут на этом шаге. Чем больше эта разность, тем выше ошибка интегрирования. Уменьшая шаг Δt_i , мы уменьшаем и локальную погрешность, рассчитанную по формуле (4.1).

1.6.2 4.2 Правило выбора шага

Если $lp_i \leq \delta_l$ — шаг успешен. Если $lp_i > \delta_l$ — шаг повторяется с уменьшенным Δt_i .

Рекомендуемое значение шага:

$$\Delta t_{i+1} = c \Delta t_i \sqrt{\frac{\delta_l}{lp_i}}, \quad c < 1 \quad (4.4)$$

1.6.3 4.3 Пример расчёта с управлением шага (Шаг 2)

Выполним второй шаг интегрирования — от t_1 до t_2 . Начальные условия — результат шага 1 (раздел 3.3).

Исходные данные (после шага 1):

$$x_1 = 2.96 \times 10^{-5} \text{ м}, \quad \nu_1 = 0.05913 \text{ м/с}, \quad a_1 = 59.13 \text{ м/с}^2$$

$$\Delta t_2 = 0.001 \text{ с}, \quad \delta_l = 0.001$$

1. Прогноз скорости (явная формула):

$$\nu_2^p = \nu_1 + a_1 \Delta t_2 = 0.05913 + 59.13 \cdot 0.001 = 0.11826 \text{ м/с}$$

2. Итерационное уточнение (неявная схема):

После решения методом Ньютона (аналогично шагу 1) получаем:

$$\nu_2^c = 0.11159 \text{ м/с}$$

3. Оценка локальной погрешности (см. раздел 4.1):

$$lp_2 = \left| \frac{\nu_2^p - \nu_2^c}{2} \right| = \left| \frac{0.11826 - 0.11159}{2} \right| = 0.00333$$

4. Сравнение с допустимой погрешностью:

$$lp_2 = 0.00333 > \delta_l = 0.001 \quad \Rightarrow \quad \text{шаг необходимо уменьшить}$$

5. Расчёт нового шага (см. раздел 4.2, $c = 0.8$):

$$\Delta t_2^{(\text{новый})} = 0.8 \cdot 0.001 \cdot \sqrt{\frac{0.001}{0.00333}} = 0.438 \times 10^{-3} \text{ с}$$

6. Повторный расчёт шага 2 с уменьшенным шагом:

После повторного решения методом Ньютона:

$$\nu_2 = 0.08509 \text{ м/с}, \quad a_2 = \frac{\nu_2 - \nu_1}{\Delta t_2} = 59.21 \text{ м/с}^2, \quad x_2 = x_1 + \frac{\nu_1 + \nu_2}{2} \Delta t_2 = 6.12 \times 10^{-5} \text{ м}$$

$$lp_2 = 0.00064 \leq \delta_l \quad \Rightarrow \quad \text{шаг принят}$$

Результат шага 2:

$$x_2 = 6.12 \times 10^{-5} \text{ м}, \quad \nu_2 = 0.08509 \text{ м/с}, \quad a_2 = 59.21 \text{ м/с}^2, \quad lp_2 = 0.00064$$

Примечание

Эти данные будут использованы как начальные условия для **шага 3** (раздел 7).

1.6.4 4.4 Основные этапы численного решения

Алгоритм численного интегрирования может быть представлен в виде следующей последовательности действий:

1. **Формирование дифференциального уравнения**, описывающего поведение системы:

$$kx + \mu \frac{dx}{dt} \left| \frac{dx}{dt} \right| + m \frac{d^2x}{dt^2} - Q \sin \frac{2\pi}{T} t = 0$$

При формировании уравнения использован второй закон Ньютона.

2. **Дискретизация** — замена производных конечно-разностными соотношениями на сетке t_i . При использовании неявной схемы второго порядка уравнения связи записываются непосредственно для текущего шага:

$$\begin{cases} kx_i + \mu \nu_i |\nu_i| + ma_i - Q \sin \frac{2\pi}{T} t_i = 0 \\ x_i = x_{i-1} + \nu_{i-1} \Delta t_i + a_i \frac{\Delta t_i^2}{2} \\ \nu_i = \nu_{i-1} + a_i \Delta t_i \end{cases}$$

Тем самым исходная задача сводится к последовательности алгебраических систем.

3. **Исключение переменных** — приведение системы к одному нелинейному уравнению относительно ν_i :

$$\alpha \nu_i |\nu_i| + \beta \nu_i + \gamma = 0$$

4. **Решение нелинейного уравнения** методом Ньютона с явным прогнозом в качестве начального приближения.
5. **Адаптивный контроль шага**:
 - оценка локальной погрешности интегрирования;
 - при превышении допустимой погрешности — повтор шага с уменьшенным Δt .
6. **Обновление переменных** — вычисление a_i и x_i по найденному ν_i с использованием уравнений связи.
7. **Коррекция шага** для следующей итерации на основе отношения допустимой и фактической погрешности.

1.7 5. Автоматическое формирование математической модели

1.7.1 5.1 Автоматическое построение нелинейного уравнения и его линеаризация

На рассмотренном примере мы обозначили канву численного решения, которой придерживается алгоритм вычислительного ядра **PRADIS**. Однако формирование дифференциального уравнения проводилось «вручную», а некоторые вопросы решались неформально (например, аналитическое определение производной $df(z)/dz$). Поэтому продолжим

рассмотрение методов **PRADIS** с выяснения принципов автоматического формирования системы дифференциальных уравнений.

Вернемся к системе на рис. 1.1. Уравнение равновесия:

$$F^c + F^y + F + F^u = 0 \quad (5.1)$$

Для каждого i -го момента времени уравнение может быть записано в виде:

$$F_i^c + F_i^y + F_i + F_i^u = 0 \quad (5.2)$$

где

$$\begin{cases} F_i^c = -Q \sin \frac{2\pi}{T} t \\ F_i^y = kx_i \\ F_i = \mu \nu_i |\nu_i| \\ F_i^u = ma_i \end{cases} \quad (5.3)$$

(Обратите внимание: знаки отличаются от (1.1), поскольку в **PRADIS** при рассмотрении условий равновесия суммируются усилия, действующие со стороны системы на элементы.)

Связь между x_i , ν_i , a_i задается формулами метода интегрирования (1.7), которые воспроизведем еще раз:

$$x_i = x_{i-1} + \nu_{i-1} \Delta t_i + a_i \frac{\Delta t_i^2}{2} \quad (5.4)$$

$$\nu_i = \nu_{i-1} + a_i \Delta t_i \quad (5.5)$$

Соотношение (5.2) представляет собой алгебраическое нелинейное уравнение, которое можно решать методом Ньютона. Имеем:

$$f(z) = 0, \quad (5.6)$$

где

$$f(z) = F_i^c + F_i^y + F_i + F_i^u$$

В качестве переменной z может быть выбрана любая из компонент x_i , ν_i или a_i , поскольку они связаны соотношениями (5.4). Примем, как и ранее, $z = \nu_i$.

На каждой итерации необходимо вычислить $f(z)$ и $df(z)/dz$.

Вычисление $f(z)$ сводится к суммированию текущих значений сил при текущих значениях x_i , ν_i , a_i (i — номер шага по времени, j — номер итерации по Ньютону). Вычисляемое значение $f(z)$ представляет собой погрешность выполнения условия равновесия, которую необходимо уменьшить до допустимых пределов.

Производная $df(z)/dz$:

$$\frac{df(z)}{dz} = \frac{dF_i^c}{dz} + \frac{dF_i^y}{dz} + \frac{dF_i}{dz} + \frac{dF_i^u}{dz} \quad (5.7)$$

Каждую из производных следует представить как производную сложной функции. Для силы F_i^c :

$$\frac{dF_i^c}{dz} = \frac{\partial F_i^c}{\partial x_i} \frac{dx_i}{dz} + \frac{\partial F_i^c}{\partial \nu_i} \frac{d\nu_i}{dz} + \frac{\partial F_i^c}{\partial a_i} \frac{da_i}{dz}$$

Поскольку $z = \nu_i$:

$$\frac{dF_i^c}{dz} = \frac{\partial F_i^c}{\partial x_i} \frac{dx_i}{d\nu_i} + \frac{\partial F_i^c}{\partial \nu_i} \frac{d\nu_i}{d\nu_i} + \frac{\partial F_i^c}{\partial a_i} \frac{da_i}{d\nu_i} \quad (5.8)$$

Аналогично для остальных сил (5.9):

$$\begin{aligned} \frac{dF_i^y}{dz} &= \frac{\partial F_i^y}{\partial x_i} \frac{dx_i}{d\nu_i} + \frac{\partial F_i^y}{\partial \nu_i} \frac{d\nu_i}{d\nu_i} + \frac{\partial F_i^y}{\partial a_i} \frac{da_i}{d\nu_i} \\ \frac{dF_i}{dz} &= \frac{\partial F_i}{\partial x_i} \frac{dx_i}{d\nu_i} + \frac{\partial F_i}{\partial \nu_i} \frac{d\nu_i}{d\nu_i} + \frac{\partial F_i}{\partial a_i} \frac{da_i}{d\nu_i} \\ \frac{dF_i^u}{dz} &= \frac{\partial F_i^u}{\partial x_i} \frac{dx_i}{d\nu_i} + \frac{\partial F_i^u}{\partial \nu_i} \frac{d\nu_i}{d\nu_i} + \frac{\partial F_i^u}{\partial a_i} \frac{da_i}{d\nu_i} \end{aligned}$$

Используем уравнения связи (5.4) для получения зависимостей a_i и x_i от ν_i :

$$\begin{aligned} a_i &= \frac{\nu_i - \nu_{i-1}}{\Delta t_i} \\ x_i &= x_{i-1} + \frac{\nu_i + \nu_{i-1}}{2} \Delta t_i \end{aligned} \quad (5.10)$$

Дифференцируем (5.10) по ν_i :

$$\frac{dx_i}{d\nu_i} = \frac{\Delta t_i}{2}, \quad \frac{d\nu_i}{d\nu_i} = 1, \quad \frac{da_i}{d\nu_i} = \frac{1}{\Delta t_i} \quad (5.11)$$

1.7.2 5.2 Вычисление частных производных

С использованием зависимостей (5.3) вычислим частные производные, входящие в выражения (5.8) и (5.9).

Для силы F_i^c :

$$\frac{\partial F_i^c}{\partial x_i} = 0, \quad \frac{\partial F_i^c}{\partial \nu_i} = 0, \quad \frac{\partial F_i^c}{\partial a_i} = 0, \quad (5.12)$$

поскольку F_i^c не зависит от перемещения, скорости и ускорения тела (5.12).

Для силы F_i^y :

$$\frac{\partial F_i^y}{\partial x_i} = k, \quad \frac{\partial F_i^y}{\partial \nu_i} = 0, \quad \frac{\partial F_i^y}{\partial a_i} = 0 \quad (5.13)$$

Для силы F_i :

$$\frac{\partial F_i}{\partial x_i} = 0, \quad \frac{\partial F_i}{\partial \nu_i} = 2\mu|\nu_i|, \quad \frac{\partial F_i}{\partial a_i} = 0 \quad (5.14)$$

Для силы F_i^u :

$$\frac{\partial F_i^u}{\partial x_i} = 0, \quad \frac{\partial F_i^u}{\partial \nu_i} = 0, \quad \frac{\partial F_i^u}{\partial a_i} = m \quad (5.15)$$

Подставляя найденные значения частных производных и коэффициенты из (5.11) в формулы (5.8) и (5.9), получаем:

$$\frac{\partial F_i^c}{\partial z} = 0, \quad \frac{\partial F_i^y}{\partial z} = \frac{k\Delta t}{2}, \quad \frac{\partial F_i}{\partial z} = 2\mu|\nu_i|, \quad \frac{\partial F_i^u}{\partial z} = \frac{m}{\Delta t_i} \quad (5.16)$$

Суммируя слагаемые в формуле (5.7), приходим к выражению:

$$\frac{df(z)}{dz} = \frac{k\Delta t_i}{2} + 2\mu|\nu_i| + \frac{m}{\Delta t_i} \quad (5.17)$$

Полученный результат структурно аналогичен ранее выведенной формуле (3.6), коэффициенты которой определяются соотношениями (2.6).

1.7.3 5.3 Преимущества автоматического подхода

Теперь, имея возможность вычислять $f(z)$ и $df(z)/dz$, продолжить расчет по описанному алгоритму не представляет труда. Какими же преимуществами окупаются дополнительные выкладки?

1. Дифференциальное уравнение движения выписывать не пришлось.
2. Развернутая форма нелинейного уравнения вида (2.6) оказалась невостребованной.
3. Функциональная зависимость для производной $df(z)/dz$ не потребовалась.

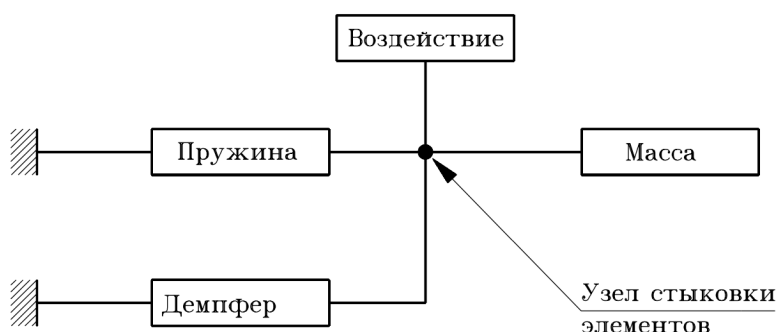


Рис. 7: Рис. 5.1

Для формирования математической модели была использована следующая информация (далее — «**перечень**»):

1. **Сведения о стыковке элементов** схемы (рис. 5.1).
2. **Условие равновесия сил** для i -го момента времени — нелинейное алгебраическое уравнение относительно x_i, ν_i, a_i :

$$\sum_{k=1}^4 F_i^{(k)} = 0 \quad (5.18)$$

(топологическое уравнение, определяемое структурой связей).

3. **Компонентные уравнения** — выражения, позволяющие определить усилия в каждом элементе как функции перемещения, скорости, ускорения и времени (5.3).
4. **Частные производные** усилий по перемещению, скорости, ускорению (5.12)–(5.15).
5. **Алгебраические уравнения связи** x, ν, a — формулы метода интегрирования (5.4).
6. **Выбор базисной переменной** z (из x, ν или a) для решения нелинейного уравнения.

Этой информации достаточно для реализации машинных алгоритмов формирования математической модели объекта.

1.8 6. Модели элементов

В качестве объекта моделирования рассматривается техническая система, расчетная схема которой изображена на рис. 1.1.

В программном комплексе Pradis модель системы строится в виде набора элементов, объединяемых в узлах. Набор степеней свободы, приписываемых элементу, зависит от его типа (рис. 6.1).

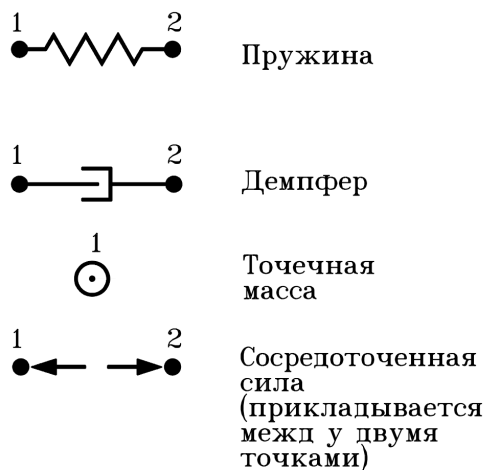
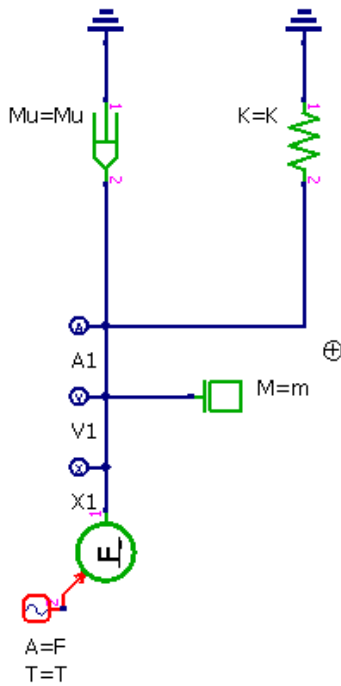


Рис. 8: Рис. 6.1

Работа строится на использовании **моделей элементов**. Из них, как из деталей конструктора, пользователь формирует общую **модель системы**. Его задача — обеспечить корректную стыковку компонентов; формирование математического описания полностью берет на себя ПО.

Для построения расчетной схемы, приведенной на рис. 1.1, используются готовые модели элементов из библиотеки base («Базовые компоненты»).



Математический маятник

Идеализированная физическая модель, представляющая собой материальную точку (тело, размерами которого можно пренебречь), подвешенную на невесомой, нерастяжимой нити и совершающую колебания в поле силы тяжести.

Начальные данные:

$k = 20000 \text{ Н/м}$
 $\mu = 1000 \text{ Н*с}^2/\text{м}^2$
 $m = 0.1 \text{ кг}$
 $Q = 1000$
 $T = 0.2 * \pi$
 $x_0 = 0$
 $v = 0$
 $dt_1 = 0.001 \text{ с}$

Рис. 9: Рис. 6.2

Этим описанием пользователь сообщает программному комплексу **PRADIS** всю необходимую информацию по стыковке элементов схемы (пункт 1 перечня п.5.3). Выбраны модели **K**, **MUNL**, **M**, **FSIN**; они соединены надлежащим образом; узел **1** описан как неподвижный.

В процессе обработки описания структуры определяется размерность системы уравнений — количество узлов, в которых должны выполняться условия равновесия. В рассматриваемом примере два узла, один из которых закреплен. На этапе формирования математической модели структура данных готовится по обоим узлам, но на этапе расчета уравнение для закрепленного узла исключается, а его кинематические характеристики обнуляются.

Этап численного интегрирования представляет собой последовательность шагов по времени, каждый из которых сводится к решению нелинейного уравнения равновесия вида (5.18). Для этого необходима информация, приведенная в пунктах 3–6 перечня (раздел 5.3).

Входными данными для любой модели элемента являются:

- неизменный список параметров модели;
- текущие значения перемещений, скоростей и ускорений узлов, с которыми элемент соединен.

По этим данным модель элемента обязана для текущего момента времени вычислить:

- 1) усилия, действующие со стороны системы на элементы (вектор усилий элемента) — пункт 3 перечня (раздел 5.3);
- 2) частные производные усилий по перемещениям, скоростям и ускорениям узлов элемента (матрицу Якоби элемента) — пункт 4 перечня (раздел 5.3).

Если элемент имеет N степеней свободы, длина вектора усилий элемента равна N , а якобиан элемента имеет размер $N \times N \times 3$.

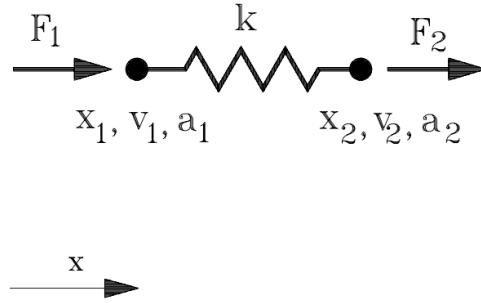


Рис. 10: Рис. 6.3

Например, разработчик двухузловой модели одномерной упругой пружины реализовал следующие зависимости (рис. 6.3):

$$F_1 = k(x_1 - x_2), \quad F_2 = k(x_2 - x_1) \quad (6.1)$$

$$\frac{\partial F_1}{\partial x_1} = k, \quad \frac{\partial F_1}{\partial x_2} = -k, \quad \frac{\partial F_2}{\partial x_1} = -k, \quad \frac{\partial F_2}{\partial x_2} = k \quad (6.2)$$

$$\frac{\partial F_1}{\partial v_1} = \frac{\partial F_1}{\partial v_2} = \frac{\partial F_2}{\partial v_1} = \frac{\partial F_2}{\partial v_2} = 0 \quad (6.3)$$

$$\frac{\partial F_1}{\partial a_1} = \frac{\partial F_1}{\partial a_2} = \frac{\partial F_2}{\partial a_1} = \frac{\partial F_2}{\partial a_2} = 0 \quad (6.4)$$

Для любого момента времени модель элемента по переданным значениям перемещений, скоростей и ускорений вычисляет усилия на концах пружины и частные производные. Вектор усилий состоит из двух элементов, якобиан — из 12.

Поскольку узел 1 пружины закреплен, востребуется только информация, связанная с незакрепленным вторым узлом:

$$F_2 = k(x_2 - x_1) \quad (6.5)$$

$$\frac{\partial F_2}{\partial x_2} = k, \quad \frac{\partial F_2}{\partial v_2} = \frac{\partial F_2}{\partial a_2} = 0 \quad (6.6)$$

Эта информация позволяет учесть вклад пружины при решении нелинейного уравнения (5.6). Вклад остальных элементов учитывается аналогично.

1.9 7. Алгоритм выполнения шага интегрирования

Продолжим ранее начатое интегрирование, сделав третий шаг по времени, используя формальный алгоритм, базирующийся на формулах (5.6)–(5.17).

По результатам второго шага ($t_2 = 1.438 \times 10^{-3}$ с):

$$x_2 = 6.12 \times 10^{-5} \text{ м}, \quad v_2 = 0.08509 \text{ м/с}, \quad a_2 = 59.21 \text{ м/с}^2$$

Величина шага $\Delta t_2 = 0.438 \times 10^{-3}$ с, локальная погрешность $lp_2 = 0.000018$.

Рекомендуемое значение Δt_3 определяем по формуле (4.4) с $c = 0.8$ и $\delta_l = 0.001$:

$$\Delta t_3 = 0.8 \cdot 0.438 \times 10^{-3} \cdot \sqrt{\frac{0.001}{0.000018}} = 2.63 \times 10^{-3} \text{ с}$$

Действуем по схеме, представленной на рис. 7.1–7.2.

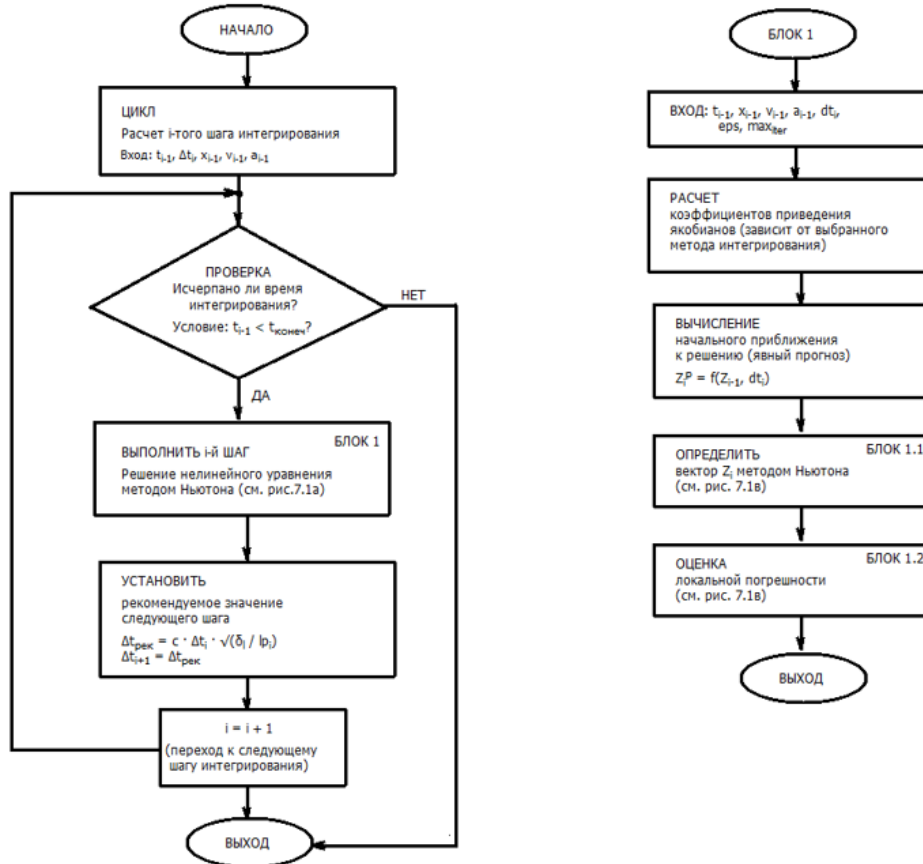


Рис. 11: Рис. 7.1

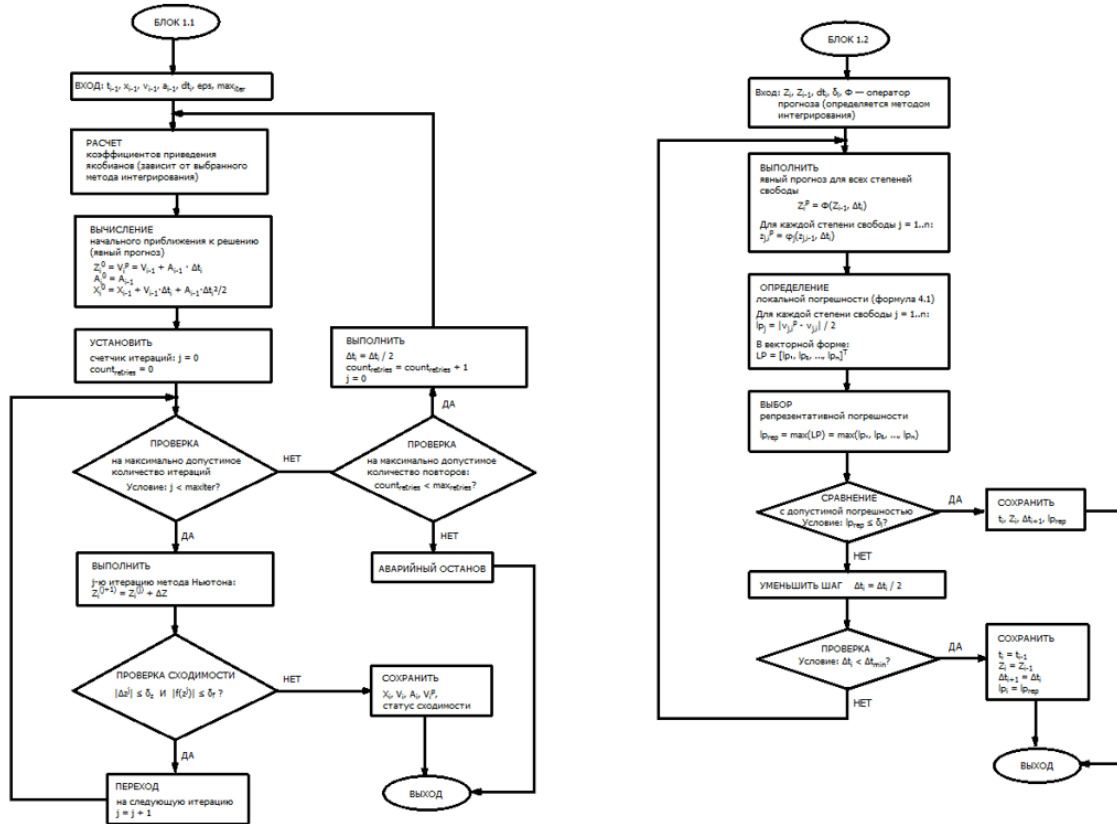


Рис. 12: Рис. 7.2

Перед третьим шагом известны значения t_2 , x_2 , v_2 , a_2 , Δt_3 .

1. **Определение коэффициентов приведения якобиана (5.11):**

$$\frac{dx_3}{dv_3} = \frac{\Delta t_3}{2} = \frac{2.63 \times 10^{-3}}{2} = 1.31 \times 10^{-3}$$

$$\frac{dv_3}{dv_3} = 1, \quad \frac{da_3}{dv_3} = \frac{1}{\Delta t_3} = \frac{1}{2.63 \times 10^{-3}} = 380.2$$

2. **Начальное приближение (явный прогноз по формуле (4.2)):**

$$v_3^0 = v_2 + a_2 \Delta t_3 = 0.08509 + 59.21 \cdot 2.63 \times 10^{-3} = 0.24081$$

Значения x_3^0 и a_3^0 , необходимые для расчета в моделях элементов:

$$a_3^0 = a_2 = 59.21$$

$$x_3^0 = x_2 + v_2 \Delta t_3 + a_3^0 \frac{\Delta t_3^2}{2} = 6.12 \times 10^{-5} + 0.08509 \cdot 2.63 \times 10^{-3} + 59.21 \cdot \frac{(2.63 \times 10^{-3})^2}{2} = 48.97 \times 10^{-5}$$

3. **Первая итерация Ньютона (см. рис. 7.1в).**

4. **Обращение к моделям элементов.** Вычисление вектора сил и якобиана каждого элемента по текущим значениям x_3^0 , v_3^0 , a_3^0 для незакрепленного узла:

Пружина:

$$F^y = kx = 20000 \cdot 46.5 \times 10^{-5} = 9.3$$

$$\frac{\partial F^y}{\partial x} = k = 20000, \quad \frac{\partial F^y}{\partial \nu} = \frac{\partial F^y}{\partial a} = 0$$

Демпфер:

$$F = \mu\nu|\nu| = 1000 \cdot 0.24081 \cdot |0.24081| = 58.0$$

$$\frac{\partial F}{\partial x} = 0, \quad \frac{\partial F}{\partial \nu} = 2\mu|\nu| = 2 \cdot 1000 \cdot 0.24081 = 481.6, \quad \frac{\partial F}{\partial a} = 0$$

Точечная масса:

$$F^u = ma = 0.1 \cdot 59.21 = 5.9$$

$$\frac{\partial F^u}{\partial x} = \frac{\partial F^u}{\partial \nu} = 0, \quad \frac{\partial F^u}{\partial a} = m = 0.1$$

Прикладываемая сила:

$$F^c = Q \sin \frac{2\pi}{T} t = -1000 \sin \frac{2\pi}{0.2\pi} (1.438 \times 10^{-3} + 2.63 \times 10^{-3}) = -40.6$$

$$\frac{\partial F^c}{\partial x} = \frac{\partial F^c}{\partial \nu} = \frac{\partial F^c}{\partial a} = 0$$

5. **Суммирование сил** (значение $f(z)$ на текущей итерации):

$$\sum F = F^y + F + F^u + F^c = -40.6 + 9.3 + 58.0 + 5.9 = 32.6$$

6. **Вычисление** $df(z)/dz$ по формулам (5.7), (5.8), (5.9):

$$\frac{dF^c}{dz} = 0 \cdot 1.31 \times 10^{-3} + 0 \cdot 1 + 0 \cdot 380.2 = 0$$

$$\frac{dF^y}{dz} = 20000 \cdot 1.31 \times 10^{-3} + 0 \cdot 1 + 0 \cdot 380.2 = 26.2$$

$$\frac{dF}{dz} = 0 \cdot 1.31 \times 10^{-3} + 481.6 \cdot 1 + 0 \cdot 380.2 = 481.6$$

$$\frac{dF^u}{dz} = 0 \cdot 1.31 \times 10^{-3} + 0 \cdot 1 + 0.1 \cdot 380.2 = 38.0$$

$$\frac{df(z)}{dz} = 0 + 26.2 + 481.6 + 38.0 = 545.8$$

7. **Приращение** Δz^1 :

$$\Delta z^1 = -\frac{f(z^0)}{f'(z^0)} = -\frac{32.6}{545.8} = -0.05973$$

8. **Очередное приближение:**

$$z^1 = z^0 + \Delta z^1 = 0.24081 - 0.05973 = 0.18108$$

9. **Уточнение** x_3^1, ν_3^1, a_3^1 по формулам (5.10):

$$\begin{aligned}\nu_3^1 &= z^1 = 0.18108 \\ a_3^1 &= \frac{\nu_3^1 - \nu_2}{\Delta t_3} = \frac{0.18108 - 0.08509}{2.63 \times 10^{-3}} = 36.5 \\ x_3^1 &= x_2 + \frac{\nu_2 + \nu_3^1}{2} \Delta t_3 = 6.12 \times 10^{-5} + \frac{0.08509 + 0.18108}{2} \cdot 2.63 \times 10^{-3} = 41.1 \times 10^{-5}\end{aligned}$$

10. **Проверка условий завершения** итераций:

$$|f(z^0)| = 32.6 > \delta_f = 0.1, \quad |\Delta z^1| = 0.05973 > \delta_z = 0.001$$

Итерации необходимо продолжить.

11. **Проверка лимита итераций:**

$$j = 1 < j_{\max} = 5$$

12. **Увеличение счетчика итераций:** $j = 2$.

13. **Вторая итерация** дает:

$$\begin{aligned}f(z^1) &= -3.6, \quad \Delta z^2 = -0.00858 \\ x_3^2 &= 39.8 \times 10^{-5}, \quad \nu_3^2 = 0.17250, \quad a_3^2 = 33.2\end{aligned}$$

Условия завершения не выполнены.

14. **Третья итерация** (последняя):

$$\begin{aligned}|f(z^2)| &= 0.07 < \delta_f, \quad |\Delta z^3| = 0.00018 < \delta_z \\ x_3^3 &= 39.8 \times 10^{-5}, \quad \nu_3^3 = 0.17232, \quad a_3^3 = 33.2\end{aligned}$$

15. **Оценка локальной погрешности:**

$$lp_3 = \left| \frac{\nu_3^p - \nu_3^c}{2} \right| = \left| \frac{0.24081 - 0.17232}{2} \right| = 0.034$$

16. **Корректировка правила выбора шага.** Практика расчетов показала, что формула (4.4) приемлема только при соотношении δ_l/lp_i вблизи единицы. При значительных отклонениях рекомендуемое значение шага часто завышено. Используем уточненное правило:

$$\Delta t_{\text{рек}} = \begin{cases} c \cdot \Delta t_i \cdot \frac{\delta_l}{lp_i}, & \text{если } \frac{\delta_l}{lp_i} < 0.25 \\ c \cdot \Delta t_i \cdot \sqrt[4]{\frac{\delta_l}{lp_i}}, & \text{если } \frac{\delta_l}{lp_i} > 7 \\ c \cdot \Delta t_i \cdot \sqrt{\frac{\delta_l}{lp_i}}, & \text{если } 0.25 \leq \frac{\delta_l}{lp_i} \leq 7 \end{cases} \quad (7.1)$$

Для нашего случая:

$$\begin{aligned}\frac{\delta_l}{lp_3} &= \frac{0.001}{0.034} = 0.03 < 0.25 \\ \Delta t_{\text{рек}} &= 0.8 \cdot 2.63 \times 10^{-3} \cdot \frac{0.001}{0.034} = 0.061 \times 10^{-3} \text{ с}\end{aligned}$$

17. **Повторный расчет** с $\Delta t_3 = 0.061 \times 10^{-3}$ с дает для момента времени $t_3 = t_2 + \Delta t_3 = 1.499 \times 10^{-3}$ с:

$$x_3 = 6.64 \times 10^{-5} \text{ м}, \quad \nu_3 = 0.08862 \text{ м/с}, \quad a_3 = 58.1 \text{ м/с}^2$$

Локальная погрешность в пределах нормы. Рекомендованное значение для следующего шага: $\Delta t_{\text{рек}} = 0.264 \times 10^{-3}$ с.

Расчет на третьем шаге завершен.

1.10 8. Разделение функций в вычислительном ядре

Основное, на что следует обратить внимание, — это разделение функций между программой интегрирования и программами реализации моделей элементов.

Программе интегрирования, работающей по алгоритму рис. 2.4а–2.4в, не важно, какие процессы интегрируются. Ее зависимость от моделей элементов сводится только к своему временному получению векторов сил и матриц якобианов. Какие свойства отдельных элементов отражает эта информация, программу интегрирования не касается.

Модели элементов, в свою очередь, имеют свой уровень независимости при четко очерченных обязанностях перед «верхами». Физические свойства отдельного элемента отражаются в компонентных уравнениях на уровне модели элемента, а программа интегрирования работает на уровне уравнений равновесия потоков.

Такое разграничение функций определяет универсальность вычислительного ядра **PRADIS** — принципиальную возможность расчета любых объектов, процессы в которых подчиняются законам равновесия потоковых переменных (равновесие сил, электрических и тепловых потоков, расходов жидкости и газа).

1.11 ПРИЛОЖЕНИЕ А — формулы методов интегрирования и коэффициенты приведения якобиана

Настоящее приложение содержит справочные формулы, используемые при переходе от дифференциальной формы уравнений движения к алгебраическим соотношениям на шаге интегрирования.

В основной части документа эти формулы применяются на примере системы с одной степенью свободы. В общем случае те же соотношения используются покомпонентно для всех активных степеней свободы расчетной схемы.

На шаге интегрирования известны значения перемещения, скорости и ускорения на предыдущем шаге:

$$x_{i-1}, \nu_{i-1}, a_{i-1}$$

и требуется определить значения на текущем шаге:

$$x_i, \nu_i, a_i$$

где:

- x — перемещение или обобщенная потенциальная переменная;
- ν — первая производная потенциальной переменной по времени;

- a — вторая производная потенциальной переменной по времени;
- $\Delta t_i = t_i - t_{i-1}$ — величина текущего шага интегрирования.

В PRADIS формулы интегрирования задают алгебраические связи между x_i , ν_i , a_i и значениями, известными с предыдущего шага. Эти связи позволяют на каждом шаге интегрирования перейти от системы нелинейных ОДУ к системе нелинейных алгебраических уравнений.

1.11.1 A.1. Метод Штермера

Для метода Штермера используются соотношения:

$$x_i = x_{i-1} + \nu_{i-1}\Delta t_i + a_i \frac{\Delta t_i^2}{2}$$

$$\nu_i = \nu_{i-1} + a_i \Delta t_i$$

Если в качестве базисной неизвестной на текущем шаге выбрана скорость $z = \nu_i$, то:

$$a_i = \frac{\nu_i - \nu_{i-1}}{\Delta t_i}$$

$$x_i = x_{i-1} + \frac{\nu_{i-1} + \nu_i}{2} \Delta t_i$$

Коэффициенты приведения производных:

$$C_x = \frac{dx_i}{d\nu_i} = \frac{\Delta t_i}{2}$$

$$C_\nu = \frac{d\nu_i}{d\nu_i} = 1$$

$$C_a = \frac{da_i}{d\nu_i} = \frac{1}{\Delta t_i}$$

1.11.2 A.2. Метод Ньюмарка

В обобщенной форме метод Ньюмарка записывается через параметры β_N и γ_N :

$$x_i = x_{i-1} + \nu_{i-1}\Delta t_i + \left(\frac{1}{2} - \beta_N\right) a_{i-1}\Delta t_i^2 + \beta_N a_i \Delta t_i^2$$

$$\nu_i = \nu_{i-1} + (1 - \gamma_N) a_{i-1} \Delta t_i + \gamma_N a_i \Delta t_i$$

При выборе базисной неизвестной $z = \nu_i$:

$$a_i = \frac{\nu_i - \nu_{i-1} - (1 - \gamma_N) a_{i-1} \Delta t_i}{\gamma_N \Delta t_i}$$

$$x_i = x_{i-1} + \nu_{i-1}\Delta t_i + \left(\frac{1}{2} - \beta_N\right) a_{i-1}\Delta t_i^2 + \beta_N a_i \Delta t_i^2$$

Коэффициенты приведения производных:

$$C_x = \frac{dx_i}{d\nu_i} = \frac{\beta_N \Delta t_i}{\gamma_N}$$

$$C_\nu = 1$$

$$C_a = \frac{da_i}{d\nu_i} = \frac{1}{\gamma_N \Delta t_i}$$

1.11.3 А.3. Неявный метод Эйлера

Для неявного метода Эйлера используются соотношения:

$$x_i = x_{i-1} + \nu_i \Delta t_i$$

$$\nu_i = \nu_{i-1} + a_i \Delta t_i$$

При выборе базисной неизвестной $z = \nu_i$:

$$a_i = \frac{\nu_i - \nu_{i-1}}{\Delta t_i}$$

$$x_i = x_{i-1} + \nu_i \Delta t_i$$

Коэффициенты приведения производных:

$$C_x = \Delta t_i$$

$$C_\nu = 1$$

$$C_a = \frac{1}{\Delta t_i}$$

1.11.4 А.4. Метод трапеций

Для метода трапеций используются соотношения:

$$x_i = x_{i-1} + \frac{\nu_{i-1} + \nu_i}{2} \Delta t_i$$

$$\nu_i = \nu_{i-1} + \frac{a_{i-1} + a_i}{2} \Delta t_i$$

При выборе базисной неизвестной $z = \nu_i$:

$$a_i = 2 \frac{\nu_i - \nu_{i-1}}{\Delta t_i} - a_{i-1}$$

$$x_i = x_{i-1} + \frac{\nu_{i-1} + \nu_i}{2} \Delta t_i$$

Коэффициенты приведения производных:

$$C_x = \frac{\Delta t_i}{2}$$

$$C_\nu = 1$$

$$C_a = \frac{2}{\Delta t_i}$$

1.11.5 А.5. Использование коэффициентов в якобиане

Модель элемента возвращает частные производные потоковой переменной F по перемещению, скорости и ускорению:

$$\frac{\partial F}{\partial x}, \frac{\partial F}{\partial \nu}, \frac{\partial F}{\partial a}$$

Программа интегрирования преобразует эти производные к производной по выбранной базисной неизвестной z . Если $z = \nu_i$, то:

$$\frac{dF}{dz} = \frac{\partial F}{\partial x} C_x + \frac{\partial F}{\partial \nu} C_\nu + \frac{\partial F}{\partial a} C_a$$

Именно эта величина используется при формировании матрицы якобиана для системы нелинейных алгебраических уравнений на текущем шаге интегрирования.

В многомерной задаче такая операция выполняется для всех активных степеней свободы и всех моделей элементов, после чего локальные вклады собираются в глобальную матрицу якобиана расчетной системы.

Управление точностью интегрирования

2.1 Управление точностью интегрирования по времени (DRLTX, DABSX)

2.1.1 Назначение параметров

Параметры **DRLTX** и **DABSX** определяют допустимую локальную погрешность (ЛП) на шаге интегрирования. Локальная погрешность оценивается для каждой степени свободы как половина разности между прогнозируемым (явный прогноз) и скорректированным (неявная коррекция) значениями потенциальной переменной (скорости, температуры, давления и т.п.).

Формула для вычисления допустимой локальной погрешности:

$$\Delta = \text{DABSX} + \text{DRLTX} \times |V_c|$$

где $|V_c|$ — абсолютное значение скорректированной потенциальной переменной на текущем шаге.

2.1.2 Физическая интерпретация

Параметр	Тип	Что контролирует	Когда доминирует
DABSX	Абсолютная	Минимальная гарантированная точность	В области малых значений $ V_c $ (скорости близки к нулю)
DRLTX	Относительная	Точность относительно текущего значения	В области больших значений $ V_c $

2.1.3 Правила выбора DABSX

Значение DABSX должно выбираться исходя из амплитуды наименьших (по абсолютной величине) потенциальных переменных, для которых требуется высокая точность.

Примеры ориентировочных значений:

Область применения	Диапазон значений переменной	Рекомендуемое DABSX
Механика (скорости)	0.001 ... 0.1 м/с	0.0001 ... 0.001
Механика (скорости)	0.1 ... 10 м/с	0.001 ... 0.01
Механика (скорости)	10 ... 1000 м/с	0.01 ... 0.1
Теплотехника (температура)	1 ... 100 °C	0.01 ... 0.1
Электротехника (напряжение)	0.1 ... 10 В	0.001 ... 0.01
Гидравлика (давление)	0.1 ... 10 МПа	0.001 ... 0.01

2.1.4 Правила выбора DRLTX

Параметр DRLTX задаёт относительную точность в долях от текущего значения переменной. Значение 0.001 соответствует относительной погрешности 0.1%.

Рекомендуемые диапазоны:

Требуемая точность	DRLTX
Оценочный (грубый) расчёт	0.01 ... 0.05
Рабочий (средний) расчёт	0.001 ... 0.005
Точный (финальный) расчёт	0.0001 ... 0.0005
Анализ затухающих процессов длительностью $> 10^4$ периодов	$1e-6$... $1e-5$

2.1.5 Взаимное влияние параметров

- Если **DABSX** $\gg$ **DRLTX** $\times |V_c|$ — доминирует абсолютная составляющая. Расчет грубый в области малых значений, но может быть излишне точным в области больших значений.
- Если **DABSX** $\ll$ **DRLTX** $\times |V_c|$ — доминирует относительная составляющая. Высокая точность в области больших значений, но возможны проблемы в области малых значений (избыточное дробление шага).
- Рекомендуемое соотношение:** $DABSX \approx DRLTX \times |V_c|_{\min}$, где $|V_c|_{\min}$ — характерное минимальное значение переменной в интересующей области процесса.

2.1.6 Влияние на вычислительные затраты

Уменьшение DRLTX или DABSX в 10 раз приводит, как правило, к увеличению количества шагов интегрирования в 2–5 раз (в зависимости от жесткости системы). Для систем с широким частотным спектром влияние может быть более существенным.

2.2 Управление шагом интегрирования (SMAX, SMIN, CONTROL, WEIGHT)

2.2.1 Назначение и область применения

В процессе численного интегрирования системы дифференциальных уравнений вычислительное ядро PRADIS использует алгоритм автоматического выбора шага интегрирования. Указанный алгоритм опирается на оценку локальной погрешности (см. подраздел 2.5), успешность сходимости метода Ньютона при решении системы нелинейных уравнений (см. подраздел 2.8), а также на рекомендации, поступающие от отдельных моделей элементов.

Для ограничения диапазона, в котором может изменяться шаг интегрирования, пользователю предоставляются ключевые параметры **SMAX** (максимально допустимый шаг) и **SMIN** (минимально допустимый шаг). Кроме того, параметры **CONTROL** и **WEIGHT** позволяют управлять поведением алгоритма в областях с высокочастотными составляющими решения, а также задавать нижнюю границу уменьшения шага по критерию локальной погрешности.

2.2.2 Параметры SMAX и SMIN

SMAX определяет верхнюю границу шага интегрирования. Независимо от оценок локальной погрешности и рекомендаций моделей элементов, шаг не может превышать значение SMAX. Параметр **SMIN** задаёт нижнюю границу шага. При попытке алгоритма уменьшить шаг ниже SMIN выдается диагностическое сообщение **R 008**, и выполнение текущей программы интегрирования прекращается.

В типовых конфигурациях значение SMAX принимается равным 0.01, что ориентировано на процессы длительностью порядка единиц секунд. Для более длительных процессов значение SMAX рекомендуется увеличивать. При этом следует учитывать, что чрезмерное увеличение SMAX может привести к ухудшению сходимости решения системы нелинейных уравнений на шаге.

Значение SMIN по умолчанию ($1e-11$) в большинстве случаев изменять не рекомендуется. Увеличение SMIN может применяться в ситуациях, когда потеря шагов по несходимости решения СнЛУ происходит исключительно из-за того, что отдельные модели элементов рекомендуют неоправданно малый шаг.

2.2.3 Параметр CONTROL

Параметр **CONTROL** выполняет две функции:

1. Задаёт нижнюю границу уменьшения шага по **критерию локальной погрешности**. Если для обеспечения заданной пользователем точности (DRLTX, DABSX) требуется шаг, меньший CONTROL, то требование к точности считается выполненным, и дальнейшее дробление шага не производится. При этом вычислительный процесс продолжается с шагом, равным CONTROL.

2. Совместно с параметром **WEIGHT** обеспечивает принудительное демпфирование (гашение) высокочастотных составляющих решения.

Важно: параметр **CONTROL** ограничивает уменьшение шага **только по критерию локальной погрешности**. По другим причинам (несходимость решения **СНЛУ**, рекомендации моделей элементов) шаг может уменьшаться ниже значения **CONTROL**, но не ниже **SMIN**.

Значение **CONTROL** по умолчанию равно 0, что означает отсутствие указанного ограничения.

2.2.4 Параметр **WEIGHT**. Механизм частотно-зависимого демпфирования

Параметр **WEIGHT** задаёт весовой коэффициент, определяющий степень подавления высокочастотных колебаний при совместном использовании с параметром **CONTROL**. Механизм демпфирования встроен непосредственно в неявную схему интегрирования и не требует дополнительных затрат на моделирование диссипативных элементов.

В документации указано эмпирическое правило:

При $WEIGHT = 1$ за один период колебаний гасятся колебания с периодом, примерно на порядок меньшим величины $CONTROL$.

Более формально: эффективное демпфирование наблюдается для колебаний, частота которых превышает величину $\sim 1 / (10 \times CONTROL)$. Увеличение **WEIGHT** приводит к более интенсивному и избирательному подавлению высокочастотной составляющей.

Для сохранения частотной характеристики фильтра при изменении **CONTROL** рекомендуется придерживаться соотношения:

$$> CONTROL^2 \times WEIGHT \approx const$$

То есть при уменьшении **CONTROL** в 10 раз значение **WEIGHT** должно быть увеличено примерно в 100 раз.

2.2.5 Влияние **CONTROL** и **WEIGHT** на вычислительные затраты и точность

В документации приведён тестовый пример, содержащий пять колебательных контуров с периодами T , $T/10$, $T/100$, $T/500$ и $T/1000$ (см. подраздел 2.7). Результаты демонстрируют:

- При отсутствии заданного **CONTROL** (или **CONTROL** = 0) программа интегрирования отслеживает все частоты. Количество шагов может составлять сотни тысяч, а время расчёта — десятки минут даже для моделей малой размерности.
- При задании **CONTROL**, ограничивающего минимальный шаг на уровне $\sim T/10000$, и **WEIGHT** = 1 количество шагов сокращается на порядок. При этом высокочастотные составляющие ($T/500$, $T/1000$) затухают в течение нескольких периодов.
- Увеличение **WEIGHT** до 50–100 при соответствующем уменьшении **CONTROL** позволяет подавить высокочастотные составляющие практически мгновенно, дополнительно сократив число шагов. Платой является некоторое увеличение погрешности для наиболее высокочастотных из интересующих пользователя гармоник.

2.2.6 Практические рекомендации по выбору значений

Цель расчёта	CONTROL	WEIGHT	Примечание
Максимальная точность; допустимы большие вычислительные затраты	0 (по умолчанию)	1 (по умолчанию)	Отслеживаются все частоты
Предварительный (оценочный) расчёт; быстрый анализ	1e-5 ... 1e-4	10 ... 100	Высокочастотные составляющие подавляются
Длительный расчёт с жёсткими ограничениями по времени	$\sim T_{\min} / 10$	100 ... 1000	$T_{\min}$ — минимальный период интересующих колебаний
Тонкая фильтрация узкой полосы частот	малый (1e-6 и менее)	большой (1000 и более)	Высокая избирательность; затраты растут

2.2.7 Ограничения и совместимость

- Значение CONTROL не может превышать SMAX. Задание $CONTROL < S_{\min}$ допустимо, но нецелесообразно: в этом случае ограничение по локальной погрешности никогда не срабатывает.
- Значение WEIGHT может быть любым неотрицательным числом. При $WEIGHT = 0$ параметр CONTROL не оказывает влияния на демпфирование.
- Параметры CONTROL и WEIGHT совместно работают только для неявных схем интегрирования ($SCHEME = 0$ или 1). Для явной схемы ($SCHEME = 2$) их использование не предусмотрено.

2.2.8 Пример диагностики

При срабатывании ограничения CONTROL в процессе расчёта в файл SYSPRINT.TXT может быть выведена следующая информация:

```
Time=.5000000 Step=.100E-07 NStep= 56
1 -> Pv=.50E+06 Vold= 1.00 Aold=.100E+12 Vp=.100E+04 Vc=1.00
ЛП превышает норму.
Переход на следующий шаг интегрирования. Выбор шага определяется
значением параметра CONTROL.
```

Данное сообщение указывает, что локальная погрешность превышает допустимую, но дальнейшее дробление шага ограничено значением CONTROL. Программа продолжает интегрирование с шагом, равным CONTROL, без аварийной остановки.

2.3 Замечания по взаимодействию CONTROL, WEIGHT и точности

При использовании CONTROL и WEIGHT следует помнить, что демпфирование высокочастотных составляющих является неотъемлемым свойством применяемых неявных схем интегрирования и **не вносит дополнительной погрешности в низкочастотную область** при корректной настройке параметров. Напротив, подавление паразитных высокочастотных колебаний, не имеющих физического смысла в рамках принятой

модели, повышает достоверность анализа интересующих пользователя низкочастотных процессов.

При возникновении сообщения **R 008** (заданный SMIN не обеспечивает требуемой точности) не следует пытаться уменьшать SMIN. Вместо этого целесообразно задать CONTROL порядка $1e-6 \dots 1e-8$ и, при необходимости, скорректировать WEIGHT в соответствии с приведённым выше соотношением.

2.4 Управление точностью решения системы нелинейных уравнений (DRLTU, DABSU, DRLTI, DABSI)

2.4.1 Общие положения

На каждом шаге интегрирования система дифференциальных уравнений преобразуется в систему нелинейных уравнений (СнЛУ) относительно приращений независимых переменных. Решение СнЛУ осуществляется итерационным методом Ньютона. Процесс считается сошедшимся, если одновременно выполнены критерии сходимости:

- по **аргументу** (приращение независимой переменной) — параметры DRLTU, DABSU;
- по **невязке** (отклонение правой части от нуля) — параметры DRLTI, DABSI.

2.4.2 Критерий сходимости по аргументу (DRLTU, DABSU)

Формула для вычисления допустимого приращения независимой переменной:

$$[dU] = DABSU + DRLTU \times |U|$$

где:

- U — текущее значение независимой переменной (скорость, температура и т.п.);
- $|U|$ — её абсолютное значение.

Назначение:

Параметр	Контролирует
DABSU	Абсолютную величину изменения переменной на последней итерации. Важен, когда U близка к нулю.
DRLTU	Относительную величину изменения переменной. Важен при больших значениях U .

Физический смысл: критерий по аргументу гарантирует, что метод Ньютона вышел на «плато» решения, где дальнейшие уточнения не меняют значение переменной более чем на заданную величину.

2.4.3 Критерий сходимости по невязке (DRLTI, DABSI)

Формула для вычисления допустимой абсолютной невязки:

$$[\Sigma F] = DABSI + DRLTI \times \Sigma |F|$$

где:

- $\Sigma |F|$ — сумма абсолютных значений потоковых переменных (сил, токов, расходов), подводимых к степени свободы от всех моделей элементов.

Назначение:

Параметр	Контролирует
DABSI	Абсолютную величину нарушения баланса (невязки). Влияет на точность выполнения законов сохранения (равновесия сил, баланса токов).
DRLTI	Относительную величину нарушения баланса относительно суммарного потока.

Физический смысл: критерий по невязке гарантирует, что для каждой степени свободы выполняется закон сохранения (например, второй закон Ньютона для механики или первый закон Кирхгофа для электрических цепей).

2.4.4 Таблица параметров точности решения СнЛУ

Параметр	Тип	Объект контроля	Значение по умолчанию	Диапазон
DABSU	Абсолютный	Аргумент	0.01	>0
DRLTU	Относительный	Аргумент	0.001	>0
DABSI	Абсолютный	Невязка	0.1	>0
DRLTI	Относительный	Невязка	0.001	>0

2.4.5 Приоритет и логика проверки

Программа интегрирования на каждой итерации проверяет оба критерия. Сходимость считается достигнутой, если для **каждой** степени свободы одновременно выполняются условия:

1. $|dU| < DABSU + DRLTU \times |U|$ (критерий по аргументу)
2. $|\Sigma F| < DABSI + DRLTI \times \Sigma |F|$ (критерий по невязке)

Если хотя бы для одной степени свободы один из критериев не выполнен, метод Ньютона выполняет следующую итерацию (при условии, что не превышено максимальное количество итераций ITR).

2.4.6 Рекомендации по выбору параметров

Сценарий	DABSU		DRLTU		DABSI		DRLTI		Обоснование
Стандартный механический расчёт	1e-3	...	1e-3		1e-2	...	1e-3		Баланс между скоростью и точностью
Расчёт с жёсткими пружинами (большие силы)	1e-4	...	1e-4		1e-1		1e-3		Ужесточение по аргументу для сходимости
Гидравлические системы (малые расходы)	1e-6	...	1e-3		1e-6	...	1e-3		Ужесточение по абсолютным значениям
Электрические схемы (малые токи)	1e-9	...	1e-3		1e-9	...	1e-3		Масштабирование под величины токов
Проблемы сходимости (потеря шагов)	Увеличить в 10 раз		1e-2		Увеличить в 10 раз		1e-2		Ослабление критериев
Финальный точный расчёт	Уменьшить в 10 раз		1e-4		Уменьшить в 10 раз		1e-4		Повышение точности

2.4.7 Взаимосвязь параметров точности СнЛУ с параметрами локальной погрешности

Если...	Связь	Рекомендация
DABSU, DRLTU слишком жёсткие	Метод Ньютона требует много итераций, шаг интегрирования не растёт	Ослабить DABSU, DRLTU в 10-100 раз
DABSU, DRLTU слишком мягкие	Накопленная погрешность интегрирования может быть неоправданно велика	Ужесточить до значений, близких к DABSX, DRLTX
DABSI, DRLTI слишком жёсткие	Сложно достичь равновесия, особенно в нелинейных задачах (контакт, трение)	Принять $DABSI \approx (10 \dots 100) \times DABSX$
DABSI, DRLTI слишком мягкие	Накопленный дисбаланс сил может исказить динамику	DABSI не должно превышать характерные действующие силы

Типовое соотношение: для большинства механических задач рекомендуется $DABSI \approx 10 \times DABSX$. Это связано с тем, что невязка сил на порядок может превышать погрешность определения скорости из-за жёсткости связей.

2.4.8 Диагностика проблем

Признаки излишней жёсткости критериев:

- В статистике работы программы наблюдается большое количество потерянных итераций по несходимости СнЛУ.
- Величина шага интегрирования систематически ниже, чем позволяют оценки локальной погрешности.
- Сообщения в отладочном выводе (DEBUG=11) показывают $Pf \gg 1$ при $Px \approx 1$.

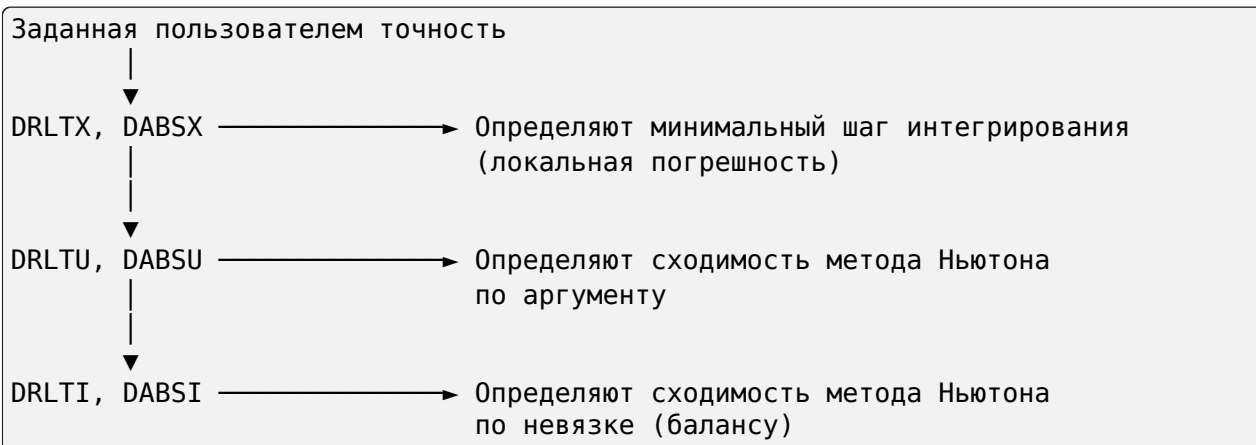
Признаки излишней мягкости критериев:

- Накопленная погрешность решения (сравнение с аналитическим или эталонным решением) недопустимо велика.

- Выходные переменные демонстрируют нефизичный «шум» или нестабильность.
- Сообщения отладочного вывода показывают, что сходимость достигается за 1-2 итерации на шаге, но результат визуально неудовлетворителен.

2.5 Взаимодействие параметров точности

2.5.1 Иерархия влияния на вычислительный процесс



2.5.2 Правило «золотого сечения»

Для большинства практических задач рекомендуется следующее соотношение между параметрами:

$$\text{DABSX} \approx \text{DABSU} / 10 \approx \text{DABSI} / 100$$

$$\text{DRLTX} \approx \text{DRLTU} \approx \text{DRLTI}$$

2.5.3 Типичные ошибки и их проявления

Ошибка	Проявление	Исправление
DABSX слишком мало по сравнению с DABSU	Потери шагов по локальной погрешности, хотя метод Ньютона сходится хорошо	Увеличить DABSX или уменьшить DABSU
DABSI слишком мало по сравнению с DABSX	Метод Ньютона не сходится даже при мелком шаге, хотя локальная погрешность мала	Увеличить DABSI в 10-100 раз
DRLTU / DRLTI слишком мягкие	Сходимость достигается за 1-2 итерации, но результат неточный	Уменьшить относительные составляющие
Все параметры слишком жёсткие	Расчёт не идёт (останов по R 006 или R 008)	Последовательно ослаблять: сначала DABSI, затем DABSU, затем DABSX