Билет №5.

1. Простейшая вариационная задача. Уравнение Эйлера. Методы решения.

Найти функцию $x^*(t)$, доставляющую экстремум функционалу вида:

$$J=\int_{t_0}^{t_1} f(t,x,x')dt \to extr$$

При этом для функции x(t) должны выполняться краевые условия $x(t_0)=x_0, x(t_1)=x_1$.

Такая задача называется *простейшей вариационной задачей.*

Для поиска x*(t) могут быть использованы необходимое и достаточное условия экстремума функционала.

Необходимым условием экстремума функционала является равенство нулю его вариаций.

Используя это условие, доказывается следующее утверждение: если для функции x*(t) выполняется необходимое условие, то функция x*(t) является решением уравнения Эйлера вида:

$$\frac{\partial f}{\partial x}-\frac{d}{dt}\left(\frac{\partial f}{\partial x'}\right)=0$$

Полученная в результате решения уравнения Эйлера функция является лишь претендентом на решение исходной вариационной задачи. Для того чтобы удостоверится, что эта функция является решением, необходима проверка достаточных условий.

Прямые методы решения вариационных задач.

Прямые методы заключаются в нахождении искомой функции, доставляющей экстремум функционалу, непосредственно ее подбором. При этом не используется необходимое условие экстремума функционала и не составляется уравнение Эйлера.

1. Метод Ритца.

В методе Ритца предложено искать решение среди линейных комбинаций заранее известных функций.

$$x=\sum_{i=0}^{n} a_i W_i(t) \qquad (**)$$

W_i - заранее известные заданные функции;

a_i - неизвестные коэффициенты.

После подстановки (**) в функционал подынтегральная функция представляет собой набор известных функций аргумента t с неизвестными коэффициентами. Интеграл может быть взят, в результате чего получим некоторую функцию $\varphi(a_0,a_1,\dots,a_n)$, для которой надо найти экстремум. Поскольку необходимо определить экстремум функции n переменных, то задача сводится от вариационной к конечномерной. Полученную задачу решают любым методом нелинейного программирования.

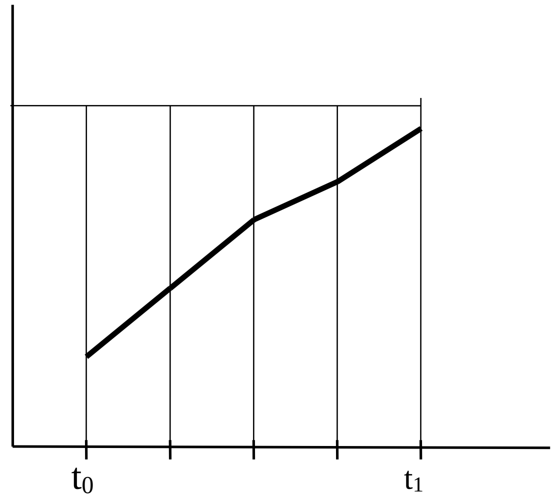
2. Метод Конторовича.

Имеет ту же основу что и метод Ритца, однако здесь допускается нелинейная относительно параметров a_i комбинация искомых функций.

Преимущества:

Может быть достигнута лучшая аппроксимация экстремали при меньшем количестве параметров a_i, в то же время более сложен вопрос о подборе функций удовлетворяющих краевым условиям.

3. Конечноразностный метод Эйлера.



Интервал $[t_0, t_1]$ разбивается на n частей. Задаются значений функции во внутренних точках отрезка. Вычисляется функционал численным методом. Меняются значения x_i любым методом нелинейного программирования, добиваясь экстремума функционала. С увеличением n решение стремится к точному.

2.Геометрические преобразования в трехмерной графике. Матрицы преобразования.

1.Перемещение.

$$T(Dx,Dy,Dz)=\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ Dx & Dy & Dz & 1 \end{vmatrix}$$

$$P' = P * T(Dx,Dy,Dz)$$
$$[x+Dx, y+Dy, z+Dz] = [x,y,z] * T(Dx,Dy,Dz)$$

2.Масштабирование.

$$S(Sx,Sy,Sz)=\begin{vmatrix} Sx & 0 & 0 & 0 \\ 0 & Sy & 0 & 0 \\ 0 & 0 & Sz & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$P' = P * S(Sx,Sy,Sz)$$
$$[x*Sx, y*Sy, z*Sz] = [x,y,z] * S(Sx,Sy,Sz)$$

Если Sx = Sy = Sz, то это однородное масштабирование.

3. Поворот.

Относительно оси OZ:

$$R_z(\alpha)=\begin{vmatrix} Cos\alpha & Sin\alpha & 0 & 0 \\ -Sin\alpha & Cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$P' = P * R_z(\alpha)$$

Относительно оси OX:

$$R_x(\alpha)=\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & Cos\alpha & Sin\alpha & 0 \\ 0 & -Sin\alpha & Cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$P' = P * R_x(\alpha)$$

Относительно оси OY:

$$R_y(\alpha)=\begin{vmatrix} Cos\alpha & 0 & Sin\alpha & 0 \\ 0 & 1 & 0 & 0 \\ Sin\alpha & 0 & Cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$P' = P * R_y(\alpha)$$

3. Напишите программу для решения уравнения методом Рунге-Кутта.

Метод Рунге-Кутта применяется для решения задачи Коши

$$\mathbf{y}' = \mathbf{f}(x, \mathbf{y}), \quad \mathbf{y}(x_0) = \mathbf{y}_0.$$

Приближенное значение в последующих точках вычисляется по итерационной формуле:

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)$$

Вычисление нового значения проходит в четыре стадии:

$$\mathbf{k}_1 = \mathbf{f}(x_n, \mathbf{y}_n),$$

$$\mathbf{k}_2 = \mathbf{f}\left(x_n + \frac{h}{2}, \mathbf{y}_n + \frac{h}{2}\mathbf{k}_1\right),$$

$$\mathbf{k}_3 = \mathbf{f}\left(x_n + \frac{h}{2}, \mathbf{y}_n + \frac{h}{2}\mathbf{k}_2\right),$$

$$\mathbf{k}_4 = \mathbf{f}(x_n + h, \mathbf{y}_n + h\ \mathbf{k}_3).$$

где h — величина шага сетки по x.

Пример. Вычислить методом Рунге-Кутта интеграл дифференциального уравнения $y' = x + y$ при начальном условии $y(0) = 1$ на отрезке [0, 0.5] с шагом интегрирования $h = 0.1$.

Решение. Вычислим y_1. Для этого сначала последовательно вычисляем k_j:

$$k_1 = x_0 + y_0$$

$$k_2 = x_0 + \frac{h}{2} + y_0 + \frac{hk_1}{2}$$

$$k_3 = x_0 + \frac{h}{2} + y_0 + \frac{hk_2}{2}$$

$$k_4 = x_0 + h + y_0 + hk_3$$

Тогда найдем y_1:

$$y_1 = y_0 + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Программа.

```
double a = 0, b = 0.5, h = 0.1;
double y0 = 1, y1 = 0;
double k1, k2, k3, k4;

for (double x = a; x < b; x += h)
{
    k1 = x + y0;
    k2 = x + h / 2 + y0 + h * k1 / 2;
    k3 = x + h / 2 + y0 + h * k2 / 2;
    k4 = x + h + y0 + h * k3;

    y1 = y0 + (k1 + 2 * k2 + 2 * k3 + k4) * h / 6;
    y0 = y1;
}
Console.WriteLine(y1);//Вывод результата
```