Необходимо создать программу для идентификации (получение модели в виде передаточной функции) на основе генетического алгоритма.

Выбирается файл (.txt, .xls), в котором в столбиках есть числовые значения параметров.
Для нескольких передаточных функций Wij выбирается, в каком столбике для этой ПФ будет входной параметр, а в каком выходной.

Сам способ задания ПФ в виде кода изображен на рисунке.
Возможны такие варианты (так как порядок числителя не может быть выше порядка знаменателя):

Вид передаточной функции	Гены, кодирующие передаточную функцию
$K\frac{1}{T_2 \cdot s+1}$	00001, 00010, 00100
$K\frac{1}{(T_2 \cdot s+1)(T_3 \cdot s+1)}$	00011, 00110, 00101
$K\frac{1}{(T_2 \cdot s+1)(T_3 \cdot s+1)(T_4 \cdot s+1)}$	00111
$K\frac{T_0 \cdot s+1}{T_2 \cdot s+1}$	10100, 10010, 10001, 01100, 01010, 01001
$K\frac{T_0 \cdot s+1}{(T_2 \cdot s+1)(T_3 \cdot s+1)}$	10110, 10101, 10011, 01110, 01101, 01011
$K\frac{T_0 \cdot s+1}{(T_2 \cdot s+1)(T_3 \cdot s+1)(T_4 \cdot s+1)}$	10111, 01111
$K\frac{(T_0 \cdot s+1)(T_1 \cdot s+1)}{(T_2 \cdot s+1)(T_3 \cdot s+1)}$	11011, 11101, 11110
$K\frac{(T_0 \cdot s+1)(T_1 \cdot s+1)}{(T_2 \cdot s+1)(T_3 \cdot s+1)(T_4 \cdot s+1)}$	11111

Все ПФ в каждой отдельной строчке, по сути, одинаковы.
То есть исключаются только гены: 10000, 01000, 11000, 11001, 11010, 11100

Совокупность передаточных функций задается в виде хромосомы (на рисунке 2).

Исходная выборка с данными (из файла) разделяется на обучающую (80%), по которой строится модель, и тестовую (20% исходных данных), по которой полученная модель сверяется.

В качестве целевой функции (функции приспособленности) предлагается использовать следующий критерий:

$$I = \sum_{i=1}^{N} \left(y_{\text{тренд.}} - y_{\text{расч.}}\right)^2, \quad (2.5)$$

где N – количество точек временного ряда;

$y_{\text{тренд.}}$ – значения из тестовой выборки;

$y_{\text{расч.}}$ – значения из тестовой выборки.

После этого, чтобы можно было вычислять непосредственно значения (для вычисления критерия), нужно перейти от передаточной функции к разностному уравнению:

Рассмотрим вывод разностного уравнения для передаточной функции

$$W_{OY}(s) = K\frac{T_0 \cdot s + 1}{(T_2 \cdot s + 1)(T_3 \cdot s + 1)}$$

Тогда, если подставить выражения в дифференциальное уравнение, можно получить выражение для y_i:

$$y_i = \frac{-\frac{T_2 T_3}{\Delta T^2} y_{i-2} + \left(\frac{2T_2 T_3}{\Delta T^2} + \frac{T_2 + T_3}{\Delta T}\right) y_{i-1} + \left(\frac{K T_0}{\Delta T} + K\right) u_i - \frac{K T_0}{\Delta T} u_{i-1}}{\frac{T_2 T_3}{\Delta T^2} + \frac{T_2 + T_3}{\Delta T} + 1}$$

y_i – значение y на текущем (i-ом) шаге функционирования модели;

y_{i-1} – значение y на предыдущем (i-1)-ом шаге функционирования модели.

u – входной параметр

ΔT – задается вручную

Передаточной функции (2.10) будет соответствовать разностное уравнение (2.11).

$$K\frac{1}{(T_2 \cdot s + 1)(T_3 \cdot s + 1)}. \quad (2.10)$$

$$y_i = \frac{K\Delta T^2 u_i - T_2 T_3 (y_{i-2} - 2y_{i-2}) + \Delta T T_2 y_{i-1} - \Delta T T_3 y_{i-1} + \Delta T^2 + K u_i}{T_2 T_3 + \Delta T T_2 + \Delta T T_3}.$$

(2.11)

Передаточной функции (2.12) будет соответствовать разностное уравнение (2.13).

$$K\frac{1}{T_2\cdot s+1}. \tag{2.12}$$

$$y_i=\frac{K\Delta Tu_i-T_2y_{i-1}+1+Ku_i}{T_2}. \tag{2.13}$$

Передаточной функции (2.14) будет соответствовать разностное уравнение (2.15).

$$K\frac{1}{(T_2\cdot s+1)(T_3\cdot s+1)(T_4\cdot s+1)}.$$

(2.14)

$$y_i=\frac{K\Delta T^3u_i-T_2T_3T_4(3y_{i-1}-3y_{i-2}+y_{i-3})+\Delta T(T_2T_3+T_2T_4+T_3T_4)(2y_{i-1}-y_{i-2})}{T_2T_3T_4+\Delta T(T_2T_3+T_2T_4+T_3T_4)+\Delta T^2(T_2+T_3+T_4)}+$$

$$+\frac{\Delta T^2(T_2+T_3+T_4)y_{i-1}-\Delta T^3+Ku_i}{T_2T_3T_4+\Delta T(T_2T_3+T_2T_4+T_3T_4)+\Delta T^2(T_2+T_3+T_4)}. \tag{2.15}$$

Передаточной функции (2.16) будет соответствовать разностное уравнение (2.17).

$$K\frac{1}{(T_2\cdot s+1)(T_3\cdot s+1)}. \tag{2.16}$$

$$y_i=\frac{K\Delta T^2u_i-T_2T_3(y_{i-2}-2y_{i-2})+\Delta TT_2y_{i-1}-\Delta TT_3y_{i-1}+\Delta T^2}{T_2T_3+\Delta TT_2+\Delta TT_3}.$$

(2.17)

Передаточной функции (2.18) будет соответствовать разностное уравнение (2.19).

$$K\frac{T_0\cdot s+1}{T_2\cdot s+1}. \tag{2.18}$$

$$y_i=\frac{K\Delta T-\Delta T+T_2y_{i-1}+KT_0(u_i-u_{i-1})}{T_2}. \tag{2.19}$$

Передаточной функции (2.14) будет соответствовать разностное уравнение (2.15).

$$K\frac{T_0\cdot s+1}{(T_2\cdot s+1)(T_3\cdot s+1)(T_4\cdot s+1)}. \tag{2.14}$$

$$y_i=$$

$$\frac{KT_0\Delta T^2(u_i-u_{i-1})+K\Delta T^3+T_2T_3T_4(3y_{i-1}-3y_{i-2}+y_{i-3})+\Delta T(T_2T_3+T_2T_4+T_3T_4)(2y_{i-1}-y_{i-2})}{T_2T_3T_4+\Delta T(T_2T_3+T_2T_4+T_3T_4)+\Delta T^2(T_2+T_3+T_4)}+$$

$$+\frac{\Delta T^2(T_2+T_3+T_4)y_{i-1}-\Delta T^3}{T_2T_3T_4+\Delta T(T_2T_3+T_2T_4+T_3T_4)+\Delta T^2(T_2+T_3+T_4)}. \tag{2.15}$$

Передаточной функции (2.14) будет соответствовать разностное уравнение (2.15).

$$K\frac{(T_0\cdot s+1)(T_1\cdot s+1)}{(T_2\cdot s+1)(T_3\cdot s+1)}. \qquad (2.14)$$

$$y_i = \frac{K(T_0T_1u_i+T_0T_1(-2u_{i-1}+u_{i-2})+\mathit{\Delta} TT_1u_i-\mathit{\Delta} TT_1u_{i-1}+\mathit{\Delta} T^2+\mathit{\Delta} TT_2u_i-\mathit{\Delta} TT_2u_{i-1})}{T_2T_3+\mathit{\Delta} TT_2+\mathit{\Delta} TT_3}. \qquad (2.15)$$

Передаточной функции (2.14) будет соответствовать разностное уравнение (2.15).

$$K\frac{(T_0\cdot s+1)(T_1\cdot s+1)}{(T_2\cdot s+1)(T_3\cdot s+1)(T_4\cdot s+1)}. \qquad (2.14)$$

$$\begin{aligned} y_i &= \frac{K(T_0T_1u_i + T_0T_1(-2u_{i-1}+u_{i-2}) + \mathit{\Delta} TT_1u_i - \mathit{\Delta} TT_1u_{i-1} + \mathit{\Delta} T^2 + \mathit{\Delta} TT_2u_i - \mathit{\Delta} TT_2u_{i-1})}{T_2T_3T_4 + \mathit{\Delta} T(T_2T_3+T_2T_4+T_3T_4) + \mathit{\Delta} T^2(T_2+T_3+T_4)} \\ &+ \frac{T_2T_3T_4(3y_{i-1} - 3y_{i-2} + y_{i-3}) + \mathit{\Delta} T(T_2T_3+T_2T_4+T_3T_4)(2y_{i-1}-y_{i-2}) + T^2(T_2+T_3+T_4)y_{i-1}}{T_2T_3T_4 + \mathit{\Delta} T(T_2T_3+T_2T_4+T_3T_4) + \mathit{\Delta} T^2(T_2+T_3+T_4)} \\ &- -\frac{\mathit{\Delta} T^3}{T_2T_3T_4 + \mathit{\Delta} T(T_2T_3+T_2T_4+T_3T_4) + \mathit{\Delta} T^2(T_2+T_3+T_4)}. \end{aligned}$$

После этого подбираются коэффициенты y(T0, T1…T4) разностного уравнения, вычисляются значения критерия, происходит отбор лучших решений и обратный переход в форму ПФ.

В итоге должны получиться передаточные функции, точно определяющие поведение параметра на тестовой выборке.

Простейший GUI приблизительно как на последней картинке.

Есть множество библиотек по ген. алгоритму для Python библиотек, например, таких как «GeneAl: a Genetic Algorithm Python Library», «pySTEP: Python Strongly Typed gEnetic Programming», «Pyvolution : Evolutionary Algorithms Framework.

Также кидаю файл с исходными данными.

1

$$W = K\frac{(T_0 \cdot s + 1)(T_1 \cdot s + 1)}{(T_2 \cdot s + 1)(T_3 \cdot s + 1)(T_4 \cdot s + 1)}$$

$$T_0 - \begin{cases} \text{присутствует } - 1, \\ \text{отсутствует } - 0 \end{cases} \qquad T_1 - \begin{cases} \text{присутствует } - 1, \\ \text{отсутствует } - 0 \end{cases} \qquad T_2 - \begin{cases} \text{присутствует } - 1, \\ \text{отсутствует } - 0 \end{cases}$$

$$T_3 - \begin{cases} \text{присутствует } - 1, \\ \text{отсутствует } - 0 \end{cases} \qquad T_4 - \begin{cases} \text{присутствует } - 1, \\ \text{отсутствует } - 0 \end{cases}$$

Рисунок 2.2 – Описание задания структуры модели в виде генетического кода

Например, передаточная функция

$$W = 3{,}45 \cdot \frac{7 \cdot s + 1}{(8 \cdot s + 1)(3 \cdot s + 1)} \qquad (2.1)$$

будет кодироваться геном 01110.

Для многосвязной системы хромосома будет состоять из генов, число которых определяется по формуле

$$i = x \times m, \tag{2.1}$$

где x – количество входных параметров;

m – количество выходных параметров.

Например, многосвязная система (рисунок 2.3), состоящая из трех входных параметров и двух выходных параметров, будет выражаться в виде хромосомы, состоящей из шести генов {(1111)(0100)(0100)(1110)(1110)(0100)}.

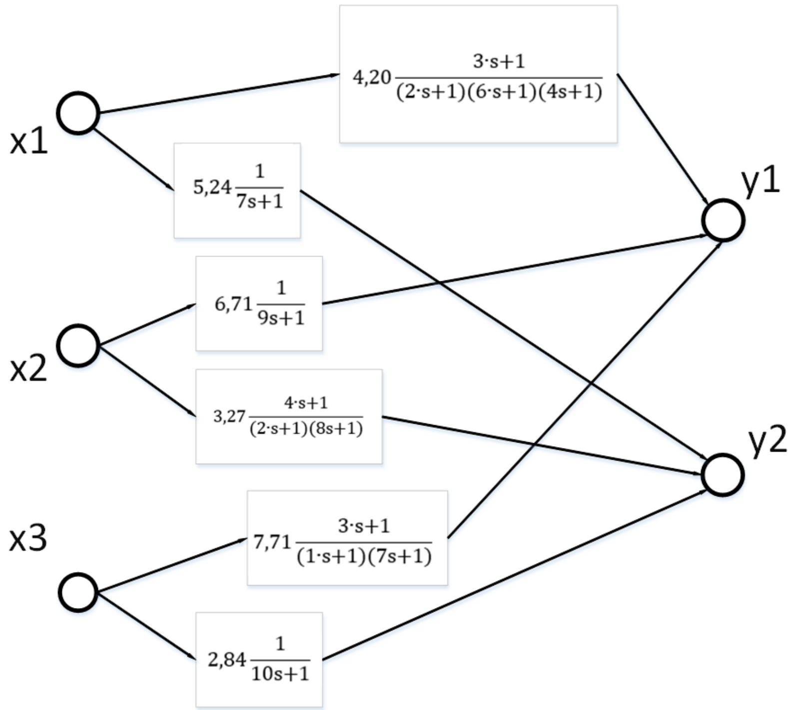


Рисунок 2.3 – Пример многосвязной системы

нетического алгоритма и описание этой блок-схемы.

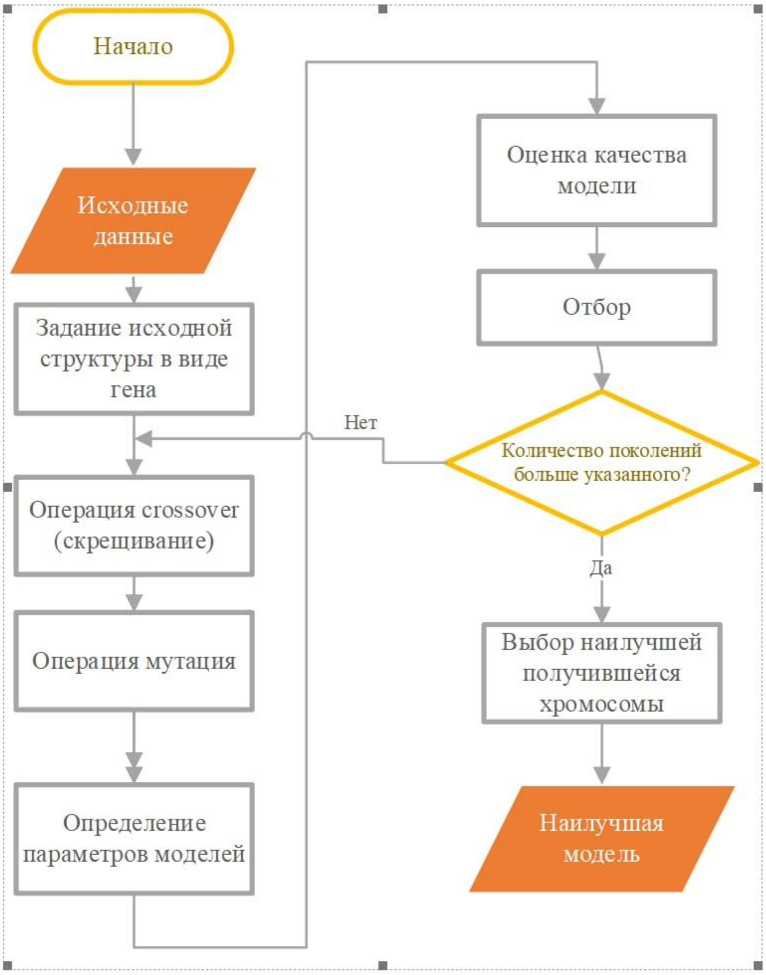
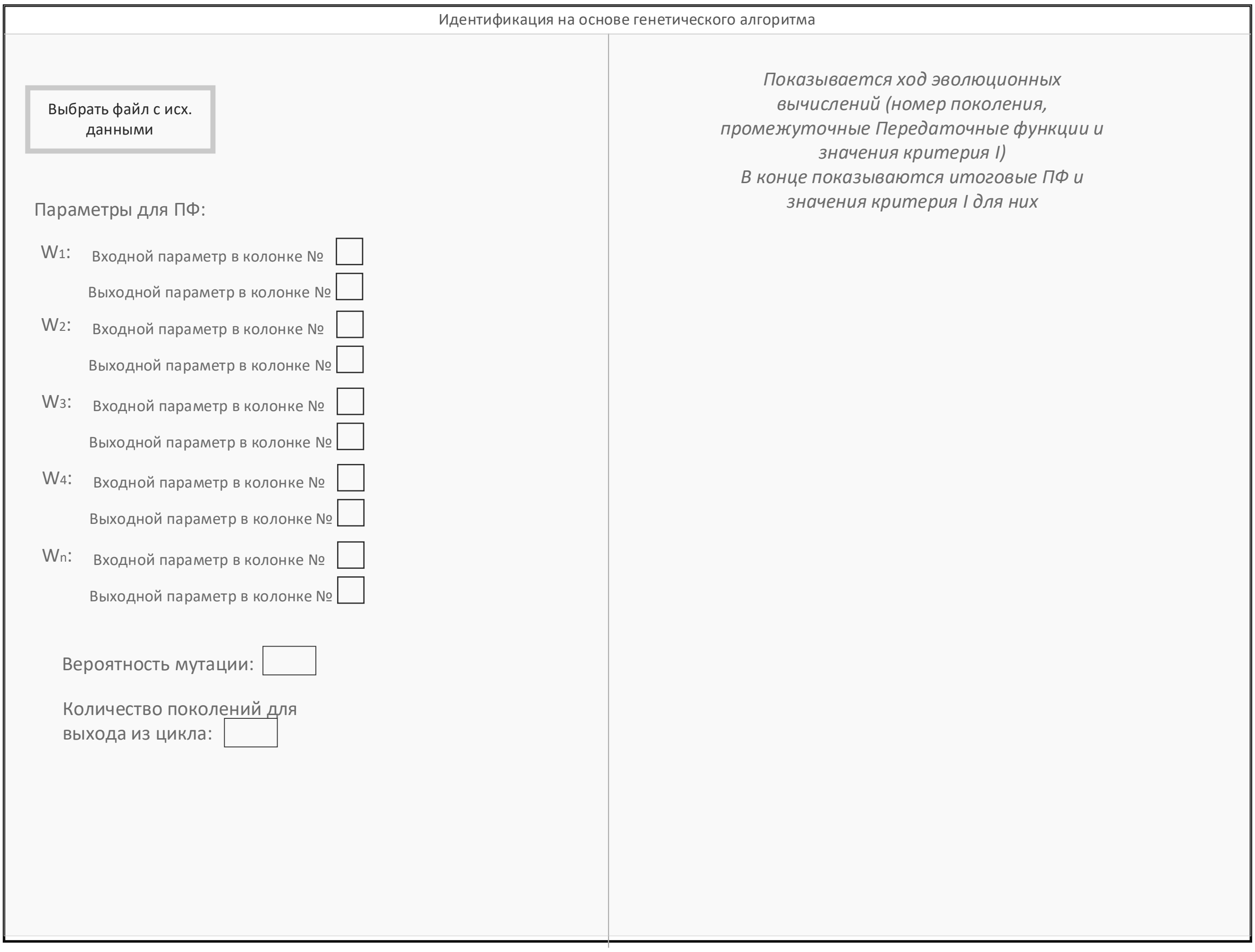


Рисунок 2.1 – Блок-схема идентификации модели с использованием генетического алгоритма

GUI



На всякий случай:

Сначала произвольно создается множество хромосом начальной популяции, причем, даже если позже выяснится, что популяция оказалась абсолютно неконкурентоспособной, генетический алгоритм все равно довольно скоро переведет ее в жизнестойкую популяцию.

Далее из полученного множества решений («поколения») выбираются решения, к которым применяются операции «кроссинговер» и «мутация».

Операция «кроссинговер» применяется для двух хромосом, которые прошли этап отбора, и служит для получения новых потомков от исходного поколения. Чаще всего выделяют следующие виды кроссинговера:

– одноточечный. В этом виде кроссинговера происходит случайный выбор точки, в которой разрываются участвующие в кроссинговере хромосомы. Родители, обмениваясь частями хромосом, образуют новых потомков (рисунок 2.4);

– двухточечный. В данном виде кроссинговера каждая хромосома разрывается в двух точках. Образование новых потомков происходит за счет обмена центральными частями родительских хромосом (рисунок 2.5);

– равномерный. В этом виде кроссинговера хромосома нового потомка образуется из родительских хромосом случайным и равновероятным образом (рисунок 2.6).

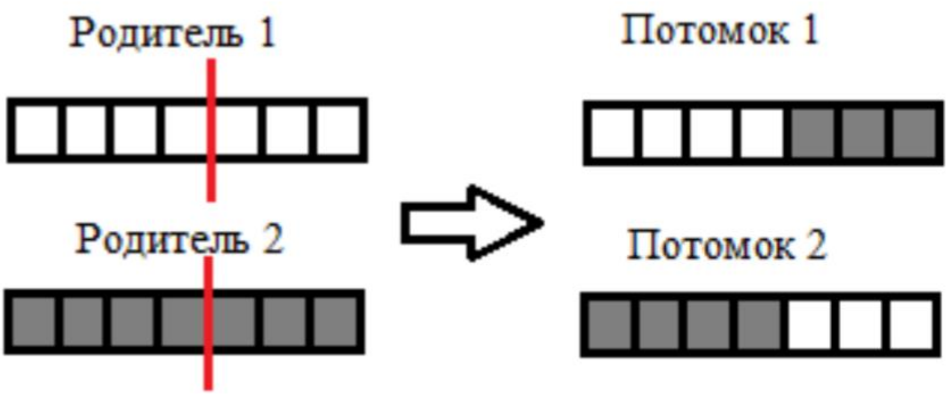


Рисунок 2.4 – Одноточечный кроссинговер

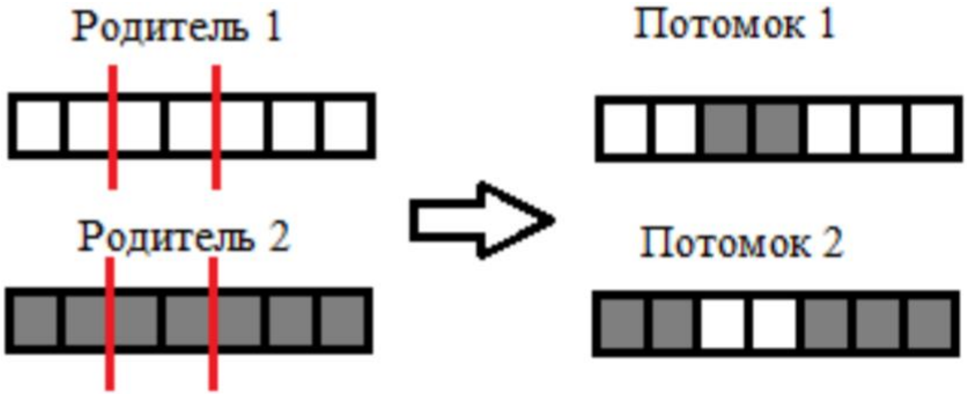


Рисунок 2.5 – Двухточечный кроссинговер

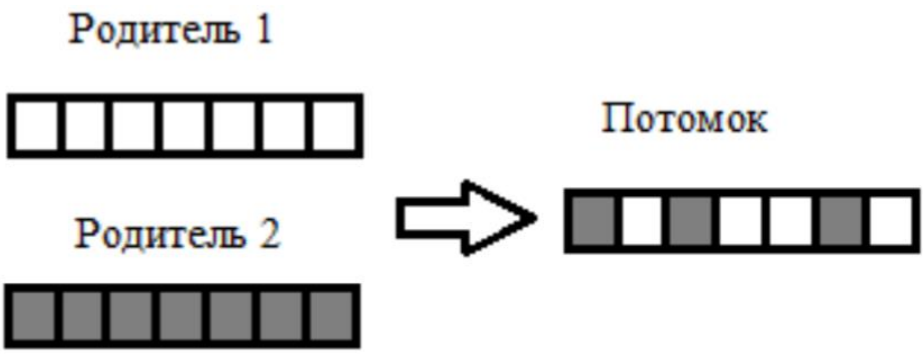


Рисунок 2.6 – Равномерный кроссинговер

После операции «кроссинговер» следует операция «мутация». Операция заключается в случайном изменении индивида, мутация позволяет избегать локальных оптимумов. Обычно выделяется три уровня мутации в зависимости от ее вероятности:

– слабый;

– средний;

– сильный.

Вероятность мутации бита для слабого уровня мутации рассчитывается по формуле (2.2):

$$p = \frac{1}{3 \cdot n}, \tag{2.2}$$

где n – общее число битов в хромосоме.

Вероятность мутации бита для среднего уровня мутации рассчитывается по формуле (2.3):

$$p = \frac{1}{n}, \tag{2.3}$$

где n – общее число битов в хромосоме.

Вероятность мутации бита для сильного уровня мутации рассчитывается по формуле (2.4):

$$p = \frac{3}{n}, \tag{2.4}$$

где n – общее число битов в хромосоме.

Операция «мутация» представлена на рисунке 2.7.

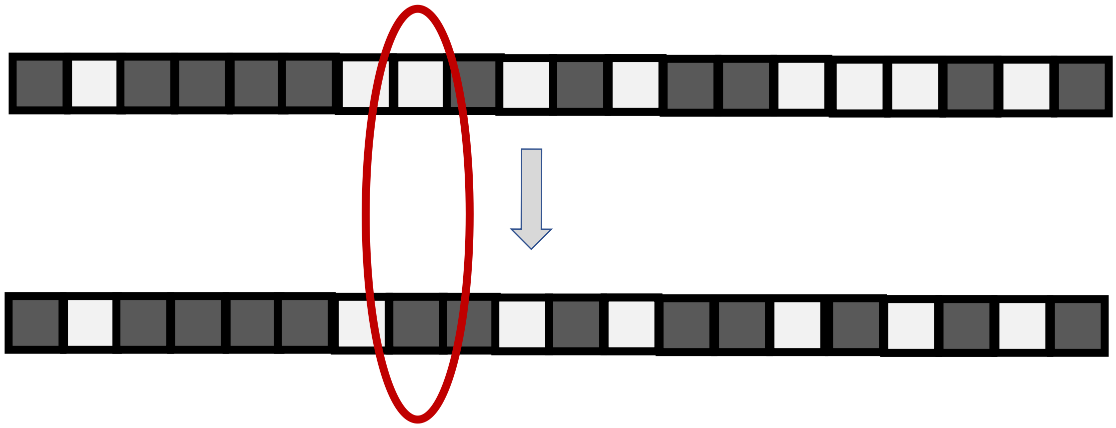


Рисунок 2.7 – Операция «мутация»

На этапе отбора нужно из всей популяции выбрать определенную ее долю, которая останется «в живых» на этом этапе эволюции.

Существует несколько схем отбора. Пропорциональный отбор (fitness proportional selection), организованный на принципе колеса рулетки считается для генетических алгоритмов основным методом отбора особей для родительской популяции с целью дальнейшего их преобразования генетическими операторами (скрещивание, мутация). Несмотря на случайность процедуры отбора отбор исходных особей происходит пропорционально значениям их функций приспособленности.

Рассмотрим метод «колеса рулетки» на примере. Допустим, имеются 5 различных хромосом с различными значениями функции приспособленности. Для каждой хромосомы вычисляется относительная пригодность по формуле

$$p_i = \frac{f(s_i)}{\sum_{i=1}^{N} f(s_i)}, \qquad (2.2)$$

где N – количество хромосом;

$f(s_i)$ – значение функции пригодности для i-ой особи;

s – хромосома.

Таблица 2.1 – Пример хромосом

Хромосома s_i	Значение функции пригодности $f(s_i)$	Значение относительной пригодности p_i
10111	4,44	0,16
11011	14,50	0,46
11100	2,12	0,08
10101	6,68	0,20
11001	3,32	0,10

Тогда на колесе рулетке площади секторов, соответствующих хромосомам, будут прямо пропорциональны (рисунок 2.4).

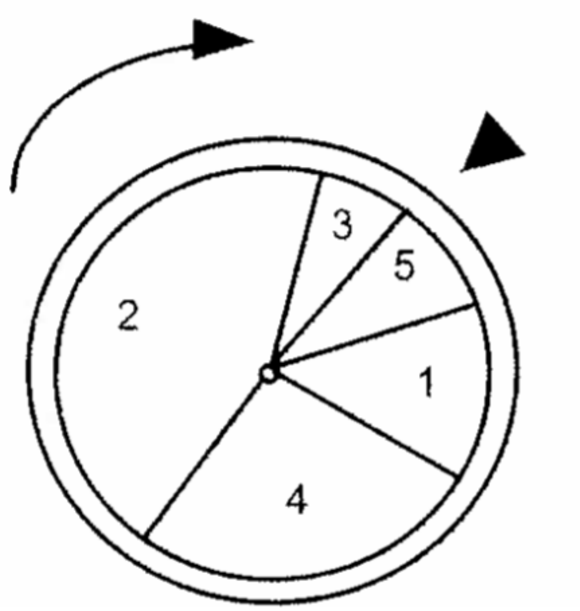


Рисунок 2.4 – Колесо рулетки

Далее колесо рулетки прокручивается N раз, выбираются N хромосом, сектор которых оказался под указателем. Очевидно, что чем больше у хромосомы значение функции пригодности, тем выше ее шансы пройти в отбор и попасть в популяцию, причем одна и та же хромосома может войти в популяцию несколько раз.

Помимо пропорционального отбора в генетическом алгоритме может также использоваться линейное ранжирование:

$$p(s_i) = \frac{1}{M}\left[\eta_{max} - (\eta_{max} - \eta_{min})\frac{i-1}{M-1}\right],$$

(2.3)

где $\eta_{min} = 2 - \eta_{max}$,

$1 \le \eta_{max} \le 2$.

Кроме вышеупомянутых схем отбора может применяться равномерное ранжирование:

$$p(s_i) = \begin{cases} \frac{1}{\mu}, \ 1 \le i \le \mu; \\ 0, \ \mu \le i \le N, \end{cases} \tag{2.4}$$

где μ – некоторое фиксированное число первых членов популяции.

Однако вышеназванные схемы отбора носят в себе определенный недостаток, заключающийся в том, что хоть и с небольшой вероятностью, но наиболее приспособленная особь в популяции может быть утрачена в каждом поколении, то есть существует вероятность того, что результаты эволюции, достигнутые на протяжении нескольких поколений, будут потеряны. Для исключения этого недостатка может использоваться так называемый элитный отбор, особенностью которого является то, что индивид с наибольшим значением функции приспособленности всегда сохраняется. При таком отборе гарантируется асимптотическая сходимость вычислений к глобальному оптимуму.

В качестве целевой функции (функции приспособленности) предлагается использовать следующий критерий:

$$I = \sum_{i=1}^{N}\left(y_{\text{тренд.}} - y_{\text{расч.}}\right)^2,$$

(2.5)

где N – количество точек временного ряда;

$y_{\text{тренд.}}$ – значения из тестовой выборки;

$y_{\text{расч.}}$ – значения из тестовой выборки.

Для вычисления параметров моделей и предлагается переходить от передаточных функций к разностным уравнениям.

Рассмотрим вывод разностного уравнения для передаточной функции $W_{OY}(s) = K\frac{T_0 \cdot s+1}{(T_2 \cdot s+1)(T_3 \cdot s+1)}$.

Тогда взаимосвязь регулируемого параметра у (выходного параметра ОУ) и управляющего воздействия u (входного параметра для ОУ) описывается операторным уравнением:

$$Y(s) \;=\; W_{OY}(s) \cdot U(s), \tag{2.6}$$

где U(s) – изображение u;

Y(s) – изображение у.

Для реализации этих соотношений в виде разностного уравнения запишем по операторному уравнению дифференциальное:

$$T_2 T_3 \frac{d^2 y}{dt^2} + (T_2 + T_3)\frac{dy}{dt} + a_0 y = K \cdot T_0 \frac{du}{dt} + K \cdot u. \qquad ?$$

Если учесть, что компьютер будет рассчитывать математическую модель дискретно через равные интервалы времени ΔT, то от дифференциалов можно перейти к разностям:

$$\frac{dy}{dt} = \frac{\Delta y}{\Delta T} = \frac{y_i - y_{i-1}}{\Delta T},$$

(2.7)

где y_i – значение у на текущем (i-ом) шаге функционирования модели;

y_{i-1} – значение у на предыдущем (i-1)-ом шаге функционирования модели.

$$\frac{d^2 y}{dt^2} = \frac{d}{dt}\left(\frac{dy}{dt}\right) = \frac{1}{\Delta T}\left(\frac{y_i - y_{i-1}}{\Delta T} - \frac{y_{i-1} - y_{i-2}}{\Delta T}\right) = \frac{y_i - 2y_{i-1} + y_{i-2}}{\Delta T^2},$$

(2.8)

где y_{i-2} – значение у на (i-2)-ом шаге функционирования модели.

Тогда, если подставить выражения в дифференциальное уравнение, можно получить выражение для y_i:

$$y_i = \frac{-\frac{T_2 T_3}{\Delta T^2} y_{i-2} + \left(\frac{2T_2 T_3}{\Delta T^2} + \frac{T_2 + T_3}{\Delta T}\right) y_{i-1} + \left(\frac{KT_0}{\Delta T} + K\right) u_i - \frac{KT_0}{\Delta T} u_{i-1}}{\frac{T_2 T_3}{\Delta T^2} + \frac{T_2 + T_3}{\Delta T} + 1}.$$

(2.9)

Аналогично выводятся разностные уравнения для других передаточных функций, структура которых задается с использованием разрабатываемой методики.

Передаточной функции (2.10) будет соответствовать разностное уравнение (2.11).

$$K\frac{1}{(T_2\cdot s+1)(T_3\cdot s+1)}. \tag{2.10}$$

$$y_i = \frac{K\Delta T^2 u_i - T_2T_3(y_{i-2}-2y_{i-2}) + \Delta T T_2 y_{i-1} - \Delta T T_3 y_{i-1} + \Delta T^2}{T_2T_3 + \Delta T T_2 + \Delta T T_3}. \tag{2.11}$$

Передаточной функции (2.12) будет соответствовать разностное уравнение (2.13).

$$K\frac{1}{T_2\cdot s+1}. \tag{2.12}$$

$$y_i = \frac{K\Delta T u_i - T_2 y_{i-1} + 1}{T_2}. \tag{2.13}$$

Передаточной функции (2.14) будет соответствовать разностное уравнение (2.15).

$$K\frac{1}{(T_2\cdot s+1)(T_3\cdot s+1)(T_4\cdot s+1)}. \tag{2.14}$$

$$y_i = \frac{K\Delta T^3 u_i - T_2T_3T_4(3y_{i-1}-3y_{i-2}+y_{i-3}) + \Delta T(T_2T_3+T_2T_4+T_3T_4)(2y_{i-1}-y_{i-2})}{T_2T_3T_4+\Delta T(T_2T_3+T_2T_4+T_3T_4)+\Delta T^2(T_2+T_3+T_4)} +$$

$$+\frac{\Delta T^2(T_2+T_3+T_4)y_{i-1} - \Delta T^3}{T_2T_3T_4+\Delta T(T_2T_3+T_2T_4+T_3T_4)+\Delta T^2(T_2+T_3+T_4)}. \tag{2.15}$$

Передаточной функции (2.16) будет соответствовать разностное уравнение (2.17).

$$K\frac{1}{(T_2\cdot s+1)(T_3\cdot s+1)}. \tag{2.16}$$

$$y_i = \frac{K\Delta T^2 u_i - T_2T_3(y_{i-2}-2y_{i-2}) + \Delta T T_2 y_{i-1} - \Delta T T_3 y_{i-1} + \Delta T^2}{T_2T_3 + \Delta T T_2 + \Delta T T_3}. \tag{2.17}$$

Передаточной функции (2.18) будет соответствовать разностное уравнение (2.19).

$$K\frac{T_0\cdot s+1}{T_2\cdot s+1}. \qquad (2.18)$$

$$y_i=\frac{K\mathit{\Delta}T-\mathit{\Delta}T+T_2y_{i-1}+KT_0(u_i-u_{i-1})}{T_2}. \qquad (2.19)$$

Передаточной функции (2.14) будет соответствовать разностное уравнение (2.15).

$$K\frac{T_0\cdot s+1}{(T_2\cdot s+1)(T_3\cdot s+1)(T_4\cdot s+1)}.$$

(2.14)

$$y_i=$$

$$\frac{KT_0\mathit{\Delta}T^2(u_i-u_{i-1})+K\mathit{\Delta}T^3+T_2T_3T_4(3y_{i-1}-3y_{i-2}+y_{i-3})+\mathit{\Delta}T(T_2T_3+T_2T_4+T_3T_4)(2y_{i-1}-y_{i-2})}{T_2T_3T_4+\mathit{\Delta}T(T_2T_3+T_2T_4+T_3T_4)+\mathit{\Delta}T^2(T_2+T_3+T_4)}+$$

$$+\frac{\mathit{\Delta}T^2(T_2+T_3+T_4)y_{i-1}-\mathit{\Delta}T^3}{T_2T_3T_4+\mathit{\Delta}T(T_2T_3+T_2T_4+T_3T_4)+\mathit{\Delta}T^2(T_2+T_3+T_4)}. \qquad (2.15)$$

Передаточной функции (2.14) будет соответствовать разностное уравнение (2.15).

$$K\frac{(T_0\cdot s+1)(T_1\cdot s+1)}{(T_2\cdot s+1)(T_3\cdot s+1)}. \qquad (2.14)$$

$$y_i=$$

$$\frac{K(T_0T_1u_i+T_0T_1(-2u_{i-1}+u_{i-2})+\mathit{\Delta}TT_1u_i-\mathit{\Delta}TT_1u_{i-1}+\mathit{\Delta}T^2+\mathit{\Delta}TT_2u_i-\mathit{\Delta}TT_2u_{i-1})}{T_2T_3+\mathit{\Delta}TT_2+\mathit{\Delta}TT_3}. \qquad (2.15)$$

Передаточной функции (2.14) будет соответствовать разностное уравнение (2.15).

$$K\frac{(T_0\cdot s+1)(T_1\cdot s+1)}{(T_2\cdot s+1)(T_3\cdot s+1)(T_4\cdot s+1)}.$$

(2.14)

$$y_i=$$

$$\frac{K\left(T_0T_1u_i+T_0T_1(-2u_{i-1}+u_{i-2})+\mathit{\Delta}TT_1u_i-\mathit{\Delta}TT_1u_{i-1}+\mathit{\Delta}T^2+\mathit{\Delta}TT_2u_i-\mathit{\Delta}TT_2u_{i-1}\right)}{T_2T_3T_4+\mathit{\Delta}T(T_2T_3+T_2T_4+T_3T_4)+\mathit{\Delta}T^2(T_2+T_3+T_4)}+$$

$$+\frac{T_2T_3T_4(3y_{i-1}-3y_{i-2}+y_{i-3})+\Delta T(T_2T_3+T_2T_4+T_3T_4)(2y_{i-1}-y_{i-2})+T^2(T_2+T_3+T_4)y_{i-1}}{T_2T_3T_4+\Delta T(T_2T_3+T_2T_4+T_3T_4)+\Delta T^2(T_2+T_3+T_4)}-$$

$$-\frac{\Delta T^3}{T_2T_3T_4+\Delta T(T_2T_3+T_2T_4+T_3T_4)+\Delta T^2(T_2+T_3+T_4)}.$$

(2.15)

Таким образом, можно выделить следующие этапы генетического алгоритма: задать целевую функцию (приспособленности) для оценки качества моделей, сгенерировать случайным образом начальную популяцию. Затем циклично проводить следующие операции над популяцией: размножение (кроссинговер), мутирование, вычисление значения критерия для данной структуры, формирование нового поколения (отбор) на основе выбора лучших в текущем. При выполнении условий остановки закончить выполнение цикла, в противном случае повторить операции.