

# К ВОПРОСУ О ПАРАМЕТРИЧЕСКОЙ ОПТИМИЗАЦИИ АЛГОРИТМОВ УПРАВЛЕНИЯ И ДИАГНОСТИКИ

В.Р. Сабанин, Н.И. Смирнов, А.И. Репин.

Московский энергетический институт (технический университет)

В статье рассматриваются пути решения многоэкстремальных задач параметрической оптимизации алгоритмов управления и диагностики. Для решения поставленных задач предлагается модифицированный генетический алгоритм (МГА). Показано, что при достаточном размере популяции (число поисковых точек) МГА с высокой вероятностью находит глобальный экстремум. В качестве примера, иллюстрирующего вычислительные возможности МГА, рассмотрен оптимальный синтез двухсвязной системы регулирования с двухканальным нейросетевым регулирующим устройством.

Получившие распространение имитационные методы исследования в задачах управления и диагностики предъявляют качественно новые требования к решению задач параметрической оптимизации. На замену классическим косвенным приемам вычисления оптимальных состояний все активнее приходят численные алгоритмы оптимизации.

Опыт исследования алгоритмов управления и диагностики показал, что для простых одноконтурных систем регулирования задачи оптимизации, как правило, являются одноэкстремальными. Однако для сложных систем управления с нейроконтроллерами и многопараметрических диагностических нейросетей характерно наличие большого числа локальных экстремумов.

Для решения одноэкстремальных задач оптимизации существует достаточное число аналитических и численных алгоритмов.

Одним из таких алгоритмов является симплекс-метод деформируемого многогранника Нелдера-Мида [1]. Универсальная программная реализация алгоритма представлена в [2]. Там же рассмотрены примеры её использования для решения различных оптимизационных задач, включая задачу параметрической идентификации математических моделей объектов управления.

При оптимизации одноконтурной АСР с ПИ-регулятором методом Нелдера-Мида было показано [3], что из различных начальных координат алгоритм осуществлял устойчивый целенаправленный поиск вектора настроечных параметров  $K_p, K_u$  для целевой функции

$$F = I_m(K_p, K_u) \cdot [1 + \alpha \cdot |\psi_{3\partial} - \psi|], \quad (1)$$

где  $I_m$  - интеграл по модулю от регулируемого параметра на временном интервале переходного процесса;  $K_p, K_u$  - настроечные параметры ПИ-регулятора;  $\psi_{3\partial}, \psi$  - соответственно, заданная и текущая степень затухания переходного процесса регулирования;  $\alpha$  - масштабный коэффициент, учитывающий вес штрафной функции.

Применение метода деформируемого многогранника для оптимизации одноконтурной системы регулирования с ПИД-регулятором, двухконтурной АСР с дифференциатором и АСР с нейрорегуляторами различной структуры привело к неоднозначности решения, результаты которого зависели от выбранных начальных координат. Из этого следует вывод о многоэкстремальности подобных задач.

Методы одноэкстремальной оптимизации не гарантировали оптимальности решения, поэтому для решения поставленных задач возникает необходимость применения методов глобальной оптимизации. В настоящее время наиболее предпочтительными методами многопараметрической многоэкстремальной оптимизации являются генетические алгоритмы [4], реализующие постулаты теории эволюции и опыта селекции растений и животных.

Стратегия поиска оптимального решения в генетических алгоритмах опирается на гипотезу селекции: чем выше приспособленность особи, тем выше вероятность того, что в потомстве, полученном с её участием, признаки, определяющие приспособленность, будут выражены ещё сильнее [5].

Генетические алгоритмы являются элементарной моделью процесса эволюции популяций особей в Природе. Если принять, что каждая особь популяции является точкой в координатном пространстве оптимизационной задачи  $X_i[x_{1i}, x_{2i}, \dots, x_{li}]$ , а приспособленность особи соответствующим значением функции цели  $f(X_i)$ , то популяцию особей можно рассматривать как множество координатных точек в пространстве, а процесс эволюции – как движение этих точек в сторону оптимальных значений целевой функции.

Подобно тому, как в природе информация о наследственных признаках организмов представлена в хромосомах в виде линейной последовательности различных комбинаций четырех нуклеотидов (А - аденин, Ц – цитозин, Т – тимин и Г – гуанин), в генетическом алгоритме векторы переменных также записываются в виде цепочек символов, используя двух-, трех- или четырехбуквенный алфавит [5]. Хромосомное представление информации позволяет применять генетические операторы: кроссовер, мутацию и инверсию. Генетические операторы являются упрощенной формой передачи наследственности. В Природе они обеспечивают процесс эволюции, а в генетических алгоритмах оптимизации движение координатных точек в направлении оптимальных значений функции цели.

Работа генетического алгоритма состоит из следующих этапов:

1. Создание в исследуемом пространстве поиска  $E^N$  начальной популяции особей размером  $\mu$  ( $\mu > N$ ), где  $N$  – размерность задачи. Исходная популяция обычно создается случайным образом сразу в символьном виде.

2. Перевод каждого вектора координат  $X_i[x_{1i}, x_{2i}, \dots, x_{li}] ; i \in \overline{0, \mu}$  из символьного вида в десятичный и расчет целевой функции для каждой координатной точки  $f_i(X_i) ; i \in \overline{0, \mu}$ .

3. Оценка популяции на вырожденность. Вырожденность популяции оценивается по разности максимального  $f^{max}$  и минимального  $f^{min}$  значений функции цели. При выполнении условия

$$|f^{max} - f^{min}| \leq \varepsilon , \quad (2)$$

где  $\varepsilon$  - достаточно малое число, популяция вырождается в точку, соответствующую решению задачи. В противном случае выполняется следующий пункт.

4. Удаление из популяции  $\rho \cdot \mu$  наименее приспособленных по значению функции цели особей, где  $\rho$  - коэффициент элиминирования (обычно он принимается равным 0.1 [6]). Оставшиеся  $(1 - \rho) \cdot \mu$  особей составляют родительскую группу, которая используется для генерации потомства (новых координатных точек).

5. Из родительской группы равновероятно  $\rho \cdot \mu$  раз выбираются пары родительских особей, к которым последовательно применяются генетические операторы. В результате генетических операций генерируются  $\rho \cdot \mu$  потомков (новых координатных точек). Полученные потомки помещаются в исходную популяцию и для них вычисляются функции цели.

6. Алгоритм переходит к п. 3 на новый этап эволюции.

Генетические алгоритмы при поиске глобального экстремума используют вероятностный подход. В связи с этим целесообразно говорить не о глобальном экстремуме, а о наилучшем достигнутом решении в принятом диапазоне поиска. Успех работы генетического алгоритма прежде всего обеспечивается идеей коллективного поиска, т.е.

поиском с помощью популяции поисковых точек и генетических операторов, заимствованных из Природы.

Генетические операторы, воздействуя с некоторой вероятностью на хромосомы родителей, обеспечивают, с одной стороны, передачу потомству информации о состоянии популяции, а с другой – поддерживают на протяжении всей эволюции достаточный уровень его изменчивости, что сохраняет поисковую способность алгоритма.

Поисковая способность генетических алгоритмов в значительной степени зависит от размера популяции  $\mu$ . Очевидно, чем больше размер популяции, тем больше вероятность приближения к искомому глобальному экстремуму. Однако на практике размер популяции  $\mu$  ограничен возможностями вычислительной техники и лежит в пределах от 10 до 500 [6].

Важной особенностью генетических алгоритмов является то, что ни один из генетических операторов (кроссовер, мутация, инверсия) в процессе генерирования потомков не опирается на знание локального рельефа поверхности целевой функции [5]. Формирование потомков происходит случайным образом, и нет гарантии, что найденные решения будут лучше родительских. Поэтому в процессе эволюции встречаются “неудачные” потомки, которые увеличивают число обращений к функции цели увеличивая, тем самым, время поиска глобального экстремума.

В настоящее время генетические алгоритмы в основном имеют специализированное применение в нейросетевых технологиях для решения многопараметрических задач распознавания образов и прогнозирования. Однако при всей внешней простоте замысла генетические алгоритмы требуют значительных усилий при настройке под конкретную задачу. В настройке нуждаются, прежде всего, вероятности применения генетических операторов.

В задачах настройки систем регулирования на детерминированные возмущения в качестве функций цели обычно выбирается интегральный критерий, вычисляемый на интервале времени переходного процесса, требующий значительного объема вычислений. Для таких задач к алгоритму оптимизации предъявляются жесткие требования по числу обращений к функции цели.

С учетом отмеченного, предлагается модификация генетического алгоритма для универсального использования в задачах небольшой размерности. Модифицированный генетический алгоритм, далее МГА, сохраняет в себе генетические качества статистической селекции популяции поисковых точек. Для исключения неудачных потомков при их генерировании в МГА реализована процедура регулярного поиска локальных экстремумов с использованием операций деформируемого многогранника.

На рис. 1 и 2 приведены две части информационных блок-схем работы МГА. В первой части реализованы процедуры статистического задания начальной популяции, сортировки и отбрасывания (элиминирования) неперспективных особей, вероятностной селекции группы особей для начала процессов регулярного поиска локальных экстремумов, а также процедур завершения работы алгоритма. Во второй части реализованы операции регулярного поиска локальных экстремумов.

В первом блоке алгоритма вводится размерность задачи оптимизации  $N$ , число особей в популяции  $\mu$ , точности решения задачи  $\epsilon$ , а также две матрицы начальных предельных максимальных и минимальных значений координат оптимизируемых векторов популяции.

В первом пункте блока 2 создается матрица случайных чисел для координат  $\mu$  начальных векторов оптимизации  $X_n$

$$xn_{i,j} = rnd(0, xn_{i,j}^{max} - xn_{i,j}^{min}) + xn_{i,j}^{min}; \quad i \in \overline{1, N}; \quad j \in \overline{1, \mu}. \quad (3)$$

Во втором пункте блока 2 для матрицы координат векторов начальной популяции вычисляются значения функции цели

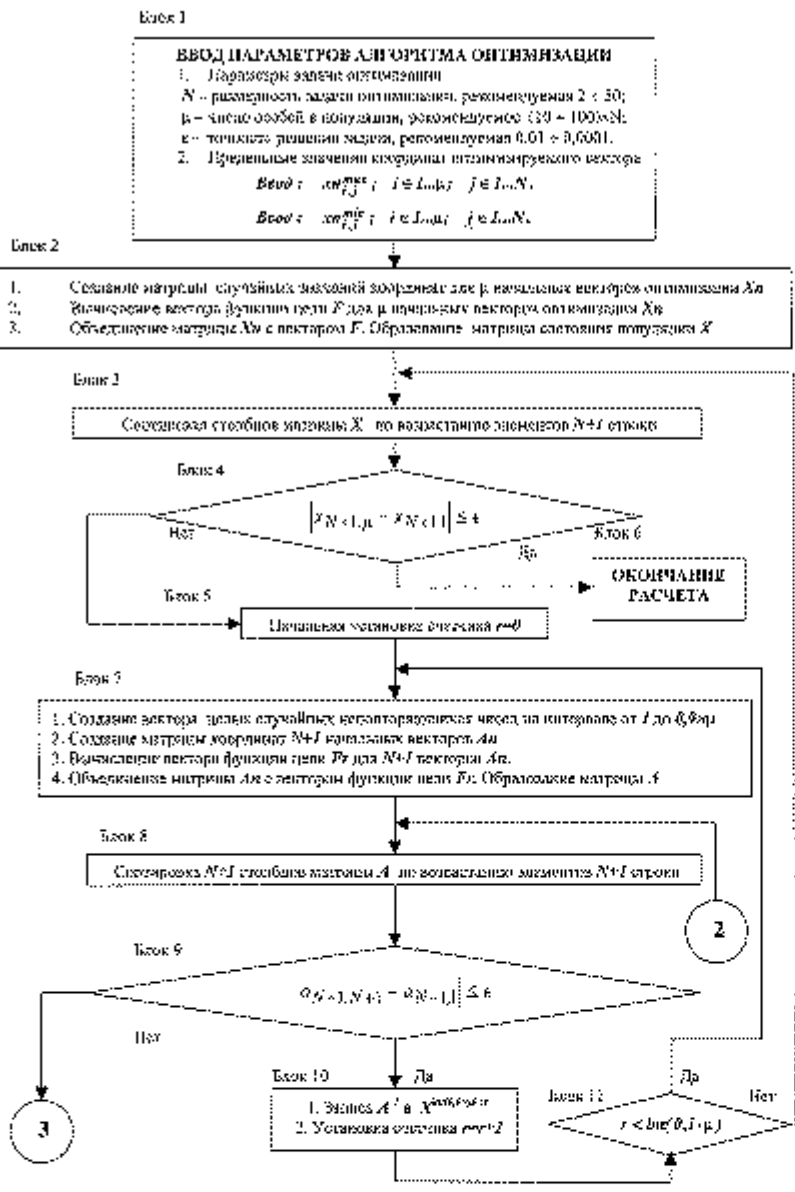


Рис. 1. Информационная блок-схема МГА (часть 1)

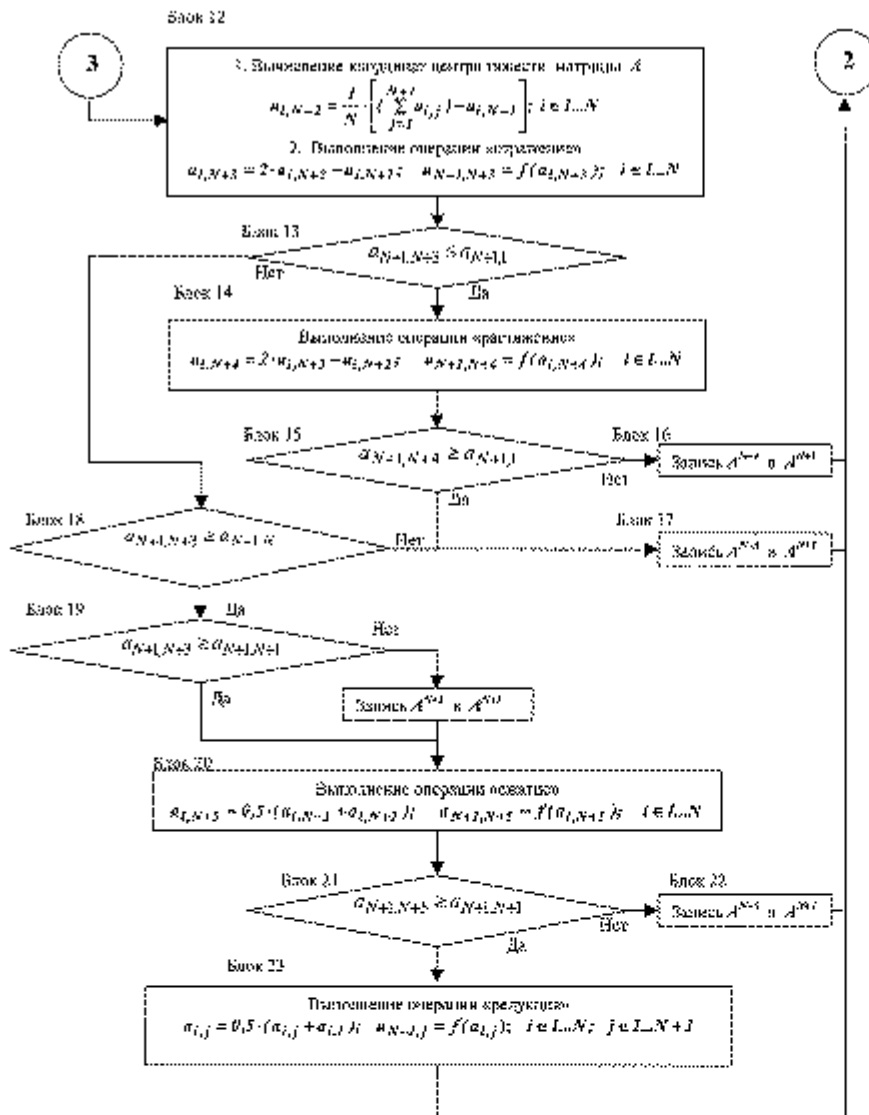


Рис. 2. Информационная блок-схема МГА (часть 2)

$$F^i = f(X_{N+1}^i); \quad i \in \overline{1, \mu}. \quad (4)$$

Функция цели задается в аналитическом виде или ее значения вычисляются численным методом по программе.

В третьем пункте блока 2 производится объединение матрицы  $X_{N+1}$  и вектора  $F$  и создается матрица состояния популяции  $X$

$$X = \begin{pmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,\mu-1} & x_{1,\mu} \\ x_{2,1} & x_{2,2} & \dots & x_{2,\mu-1} & x_{2,\mu} \\ \dots & \dots & \dots & \dots & \dots \\ x_{N,1} & x_{N,2} & \dots & x_{N,\mu-1} & x_{N,\mu} \\ x_{N+1,1} & x_{N+1,2} & \dots & x_{N+1,\mu-1} & x_{N+1,\mu} \end{pmatrix}, \quad (5)$$

где элементами последней  $N+1$  строки являются значения вектора  $F$ .

Матрица  $X$  в процессе генетического отбора обновляется от поколения к поколению и является показателем степени приближения решения к наилучшему или глобальному экстремуму.

В блоке 3 производится сортировка столбцов матрицы  $X$  по возрастанию элементов  $N+1$  строки. После сортировки последние столбцы матрицы  $X$  будут содержать неперспективные особи, которые при смене поколения популяций должны быть элиминированы, т.е. заменены на новые, более перспективные.

В блоке 4 популяция оценивается на вырожденность матрицы состояния  $X$ . Если условия блока 4 выполняются, задача считается решенной и алгоритм выходит на окончание расчета и в блоке 5 выводится первый столбец матрицы  $X$ .

В противном случае в блоке 5 устанавливается начальное значение счетчика поколений  $r=0$  и алгоритм переходит к блоку 7.

В первом пункте блока 7 создается вектор  $N+1$  целых случайных неповторяющихся чисел  $P$ . Случайные числа выбираются из интервала от 1 до  $0,9 \cdot \mu$  (по числу оставшихся в популяции перспективных особей)

$$P_j = \text{rnd}(0, 0,9 \cdot \mu); \quad j \in \overline{1, N+1} \quad (6)$$

Во втором пункте блока 7 из матрицы состояния  $X$  выбираются столбцы родительской группы  $N+1$  особей для  $r$ -го регулярного поиска потомка. Принцип формирования матрицы родительской группы  $A_r$  показан ниже

$$A_r = \begin{pmatrix} x_{1,P_1} & x_{1,P_2} & \dots & x_{1,P_N} & x_{1,P_{N+1}} \\ x_{2,P_1} & x_{2,P_2} & \dots & x_{2,P_N} & x_{2,P_{N+1}} \\ \dots & \dots & \dots & \dots & \dots \\ x_{N,P_1} & x_{N,P_2} & \dots & x_{N,P_N} & x_{N,P_{N+1}} \end{pmatrix}. \quad (7)$$

В третьем пункте блока 7 вычисляются  $N+1$  элементов вектора значений функции цели  $F_r$  для столбцов матрицы  $A_r$ , а в четвертом создается матрица состояния родительской группы особей  $r$ -го регулярного поиска локального экстремума  $A$ . Элементами  $N+1$  строки матрицы  $A$  является вектор значений функции цели  $F_r$

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,N} & a_{1,N+1} \\ a_{2,1} & a_{2,2} & \dots & a_{2,N} & a_{2,N+1} \\ \dots & \dots & \dots & \dots & \dots \\ a_{N,1} & a_{N,2} & \dots & a_{N,N} & a_{N,N+1} \\ a_{N+1,1} & a_{N+1,2} & \dots & a_{N+1,N} & a_{N+1,N+1} \end{pmatrix} \quad (8)$$

В блоке 8 производится сортировка столбцов матрицы  $A$  по возрастанию  $N+1$  строки, а в блоке 9 оценивается степень сжатия столбцов матрицы  $A$ . Оценка делается по элементам  $N+1$  строки. Если условие блока 9 выполняется, то считается, что локальный экстремум найден и в блоке 10 первый столбец матрицы  $A$  записывается в  $0,9 \times \mu + r$ -ый столбец матрицы  $X$  и устанавливается новое значение счетчика поколений  $r=r+1$ . Затем, в блоке 11 проверяется число найденных локальных экстремумов  $r$ .

Если выполняется условие

$$r < \text{int}(0,1 \cdot \mu), \quad (9)$$

где  $\text{int}(\dots)$  – целое число от аргумента, то алгоритм возвращается к блоку 3. В противном случае – к блоку 7.

Если условие (9) не выполняется, то алгоритм переходит ко второй части блок-схемы к блокам регулярного поиска локального экстремума.

В блоке 12 вычисляются координаты центра тяжести матрицы  $A$ . Центр тяжести вычисляется без  $N+1$ -го наихудшего столбца. Там же, во втором пункте выполняется операция «отражения» от наихудшего столбца и вычисляется значение функции цели для отраженного  $N+2$ -го столбца.

Если значение функции цели в отраженной точке  $a_{N+1,N+2}$  меньше или равно, чем наилучшее  $a_{N+1,1}$ , то в блоке 14 выполняется операция «растяжения» матрицы в перспективном направлении. Для нового  $a_{N+1,N+2}$  столбца вычисляется функция цели и сравнивается с наилучшим значением  $a_{N+1,1}$ . Если  $a_{N+1,N+2} < a_{N+1,1}$ , то столбец матрицы  $A^{N+2}$  записывается на место наихудшего столбца  $A^{N+1}$  и алгоритм возвращается к блоку 8 для новой сортировки столбцов матрицы  $A$ . В противном случае в столбец  $A^{N+1}$  записываются элементы столбца  $A^{N+2}$  и алгоритм также переходит к блоку 8.

Если в блоке 13  $a_{N+1,N+2} > a_{N+1,1}$ , то в блоке 18 элемент  $a_{N+1,N+2}$  сравнивается с элементом  $a_{N+1,N}$  второго после наихудшего столбца. Если  $a_{N+1,N+2} < a_{N+1,N}$ , то в столбец  $A^{N+1}$  записываются элементы столбца  $A^{N+2}$  и алгоритм возвращается к блоку 8. В противном случае функция цели в  $N+3$  столбце сравнивается с наихудшим  $N+1$  значением.

Если  $a_{N+1,N+2} < a_{N+1,N+1}$ , то в столбец  $A^{N+1}$  записываются элементы столбца  $A^{N+2}$  и алгоритм переходит к блоку 20. В противном случае алгоритм переходит к блоку 20 без передачи элементов столбца.

В блоке 20 выполняется операция «сжатия» матрицы  $A$  в пространстве между центром тяжести и наихудшим столбцом и, соответственно, вычисление функции цели в новом  $N+5$  столбце. Если  $a_{N+1,N+5} < a_{N+1,N+1}$ , то в блоке 22 в столбец  $A^{N+1}$  записываются элементы столбца  $A^{N+5}$  и алгоритм возвращается к блоку 8. В противном случае в блоке 23 происходит «редукция» матрицы  $A$ . Редукция – это уменьшение вдвое расстояния всех столбцов от наилучшего столбца  $A^1$ . Для новых значений элементов первых  $N$  строк всех столбцов матрицы  $A$  вычисляются и записываются в  $N+1$  строку значения функции цели. Затем алгоритм возвращается к блоку 8.

Для тестирования и иллюстрации работы МГА была составлена двумерная многоэкстремальная целевая функция вида

$$y(x_1, x_2) = (\sin(\pi \cdot x_1) + \sin(\pi \cdot x_2)) \cdot 0,2 + 0,01 \left[ 0,4 \cdot (x_1 - 5,5)^2 + 0,5 \cdot (x_2 - 5,5)^2 \right] + 0,4, \quad (10)$$

имеющая глобальный экстремум в точке  $(5, 5; 5, 5)$ .

На рис. 3 показаны шесть состояний поверхности отклика тестовой функции (10).

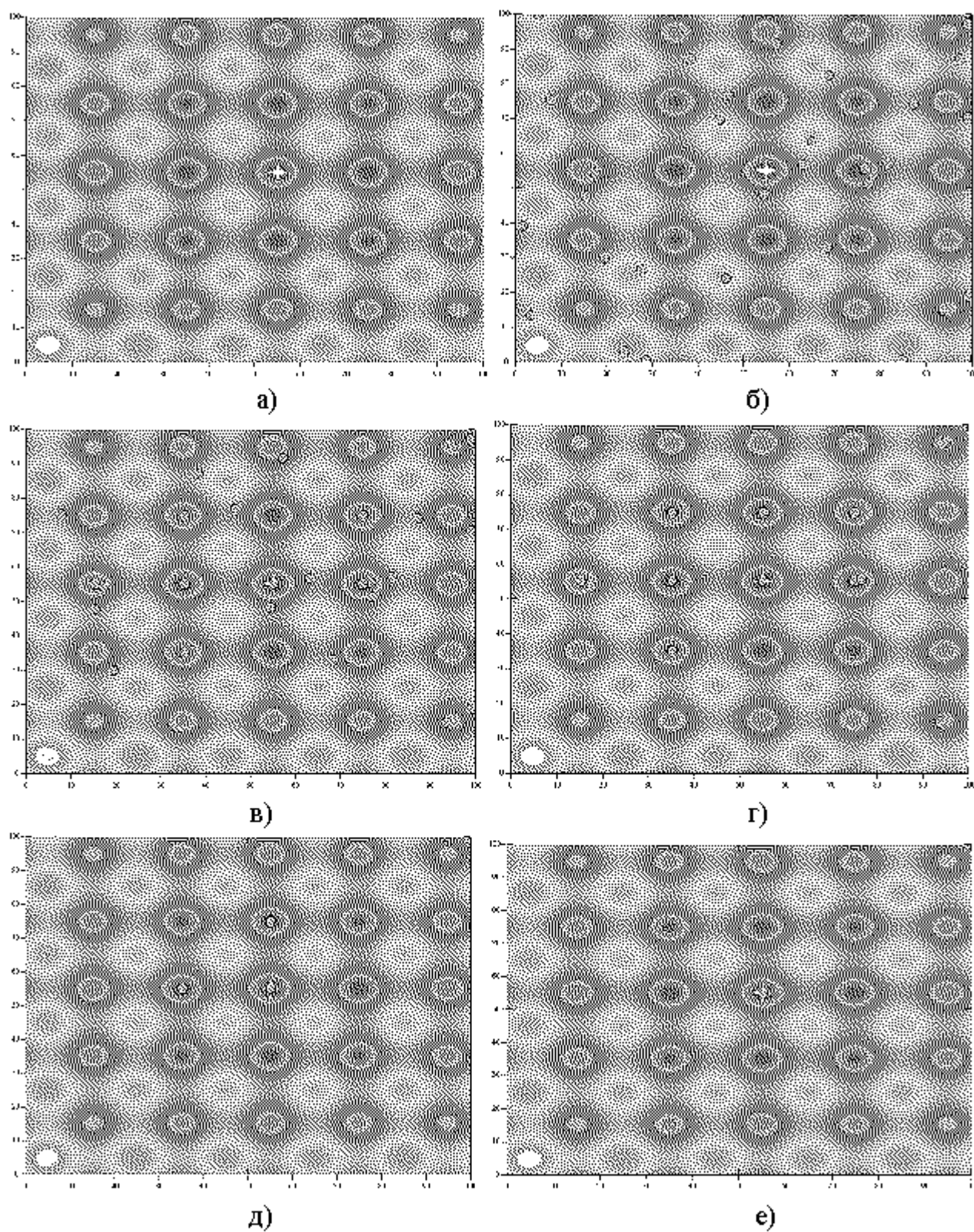


Рис.3. Поверхности отклика тестовой функции цели



На рис. 3 а звездочкой отмечена точка глобального экстремума. Состояние задачи оптимизации для различных поколений эволюции популяции показаны на рис 3 б, в, г, д, е. На рис. 3 б показан начальный разброс точек в первом поколении. В каждом поколении удалялись по 3 наихудших особи. На замену им находились точки в локальных экстремумах. Расположение точек популяции в пятом, десятом и в пятнадцатом поколении представлены соответственно на рис. 3 в, г, д. Состояние решенной задачи, когда все точки стянулись в одну точку глобального экстремума, представлено на рис 3 е. Для получения окончательного решения потребовалось семнадцать поколений.

Как было отмечено выше, задача определения глобального экстремума является вероятностной. Причем, вероятность успешного поиска в первую очередь зависит от размера популяции  $\mu$ . Результаты исследования влияния размера популяции на вероятность гарантированного определения глобального экстремума представлены на рис.4. Для каждого заданного размера популяции  $\mu$  проводилось по 1000 экспериментов, в которых определялось число неудачных решений.

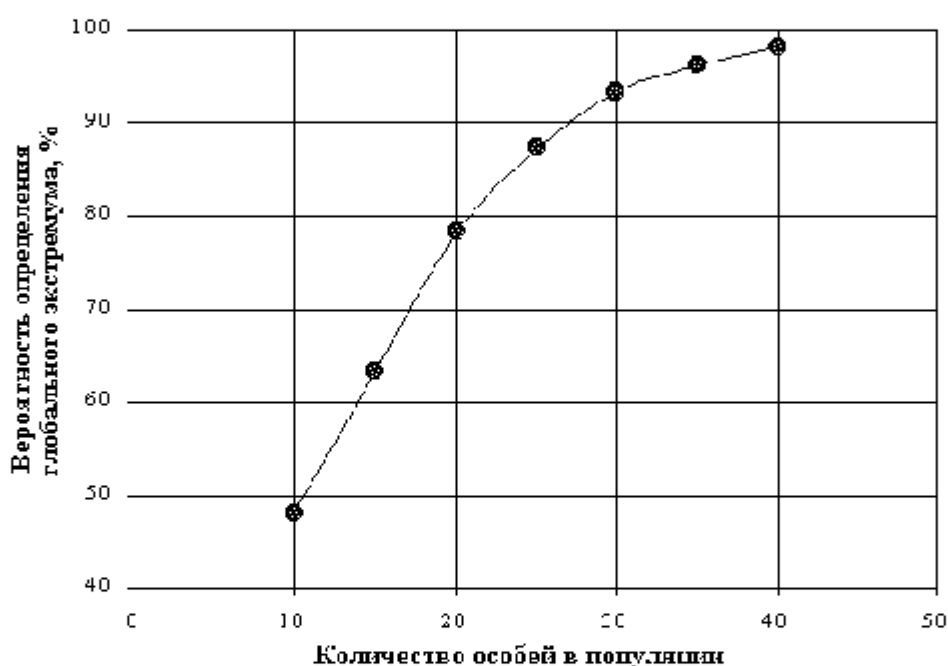


Рис.4. Результаты исследования влияния размера популяции на вероятность гарантированного определения глобального экстремума

Из графика, представленного на рис.4, видно, что при достаточном количестве особей в популяции модифицированный генетический алгоритм с высокой вероятностью находит глобальный экстремум. Так, при количестве особей в популяции  $\mu=40$  вероятность определения глобального экстремума составила величину, близкую к 100%.

В качестве примера, иллюстрирующего вычислительные возможности модифицированного генетического алгоритма, рассмотрен оптимальный синтез двухсвязной системы регулирования с многоканальным регулирующим устройством МРУ (рис.5).

Объектом регулирования является прямоточный котел с взаимосвязанными регулируемыми параметрами: давлением пара за котлом  $P_k$  с регулирующим воздействием на расход топлива  $B_m$  и температурой в промежуточной точке пароводяного тракта  $\theta_{np}$  с регулирующим воздействием на расход питательной воды  $D_{не}$ . Передаточные функции по каналам регулирования и каналам взаимных связей в соответствии с динамическими характеристиками, полученными на реальном котле [9], имеют вид:

$$W_{11}(s) = \frac{1}{(18 \cdot s + 1)^3}; \quad W_{22}(s) = -\frac{0.5}{(50 \cdot s + 1)^3};$$

$$W_{12}(s) = \frac{0.5 \cdot (23.7 \cdot s + 1)}{(18 \cdot s + 1)^3 \cdot (37 \cdot s + 1)}; \quad W_{21}(s) = \frac{0.5 \cdot (105 \cdot s + 1)}{(28 \cdot s + 1)^3 \cdot (27 \cdot s + 1)}. \quad (11)$$

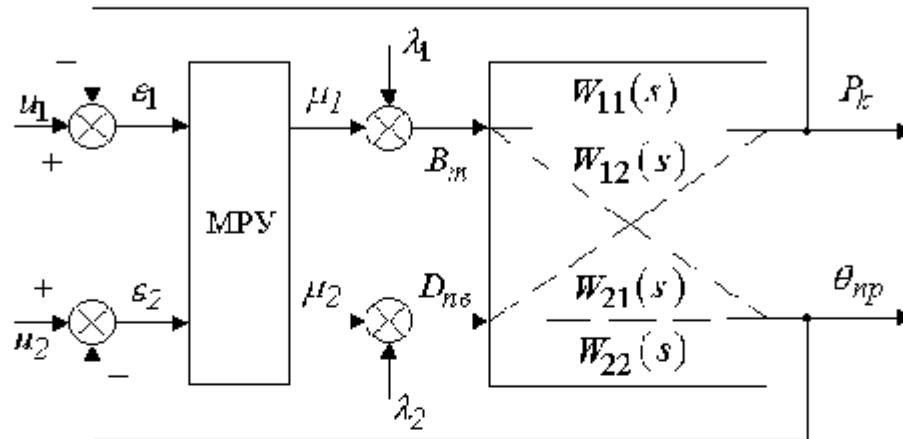


Рис. 5. Структурная схема двухсвязной АСР

Исследование проводилось методом имитационного моделирования во временной области для регулирующих устройств, реализованных следующими способами:

- с помощью двух не связанных между собой ПИ-регуляторов (4 настроечных параметра);
- с помощью нейросетевого двухканального регулятора [8] с взаимными связями между каналами (12 настроечных параметров).

Структура нейросети для реализации нейросетевого регулятора представлена на рис. 6.

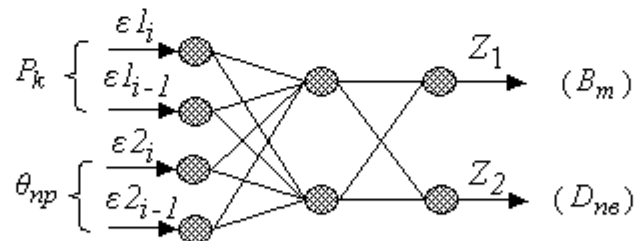


Рис. 6. Структура нейросети двухканального регулятора

Параметрическая оптимизация анализируемой двухсвязной АСР для рассматриваемых вариантов реализации регулирующего устройства проводилась с помощью модифицированного генетического алгоритма.

Переходные процессы для  $P_k$  и  $\theta_{нр}$  при найденных настроечных параметрах ПИ-регуляторов (первый способ) и оптимальных параметрах нейросети (второй способ) представлены на рис 7 а, б (соответственно, графики 1, 2). Для сравнения там же показаны переходные процессы этих параметров в АСР с ПИ-регуляторами, настройки которых найдены аналитическим методом по эквивалентным объектам, учитывающим взаимные связи (графики 3).

На рис. 7 в, г переходные процессы в анализируемой АСР с оптимально настроенным двухканальным нейросетевым регулятором (графики 2) сравниваются с процессами, полученными в работе [10] для регулирующего устройства с двумя ПИ-регуляторами и динамическим компенсатором “топливо  $B_m$  -вода  $D_{не}$ ” в виде РД-звена, параметры которых определялись с помощью алгоритма автоматизированной настройки (графики 4).

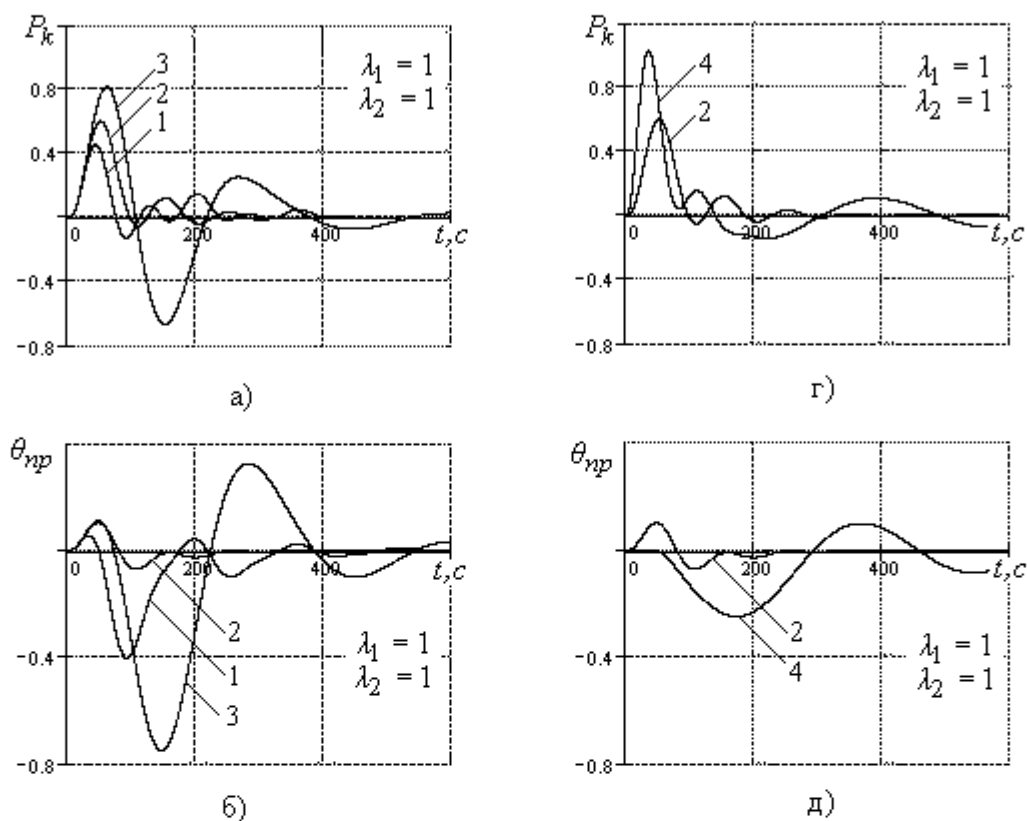


Рис.7. Переходные процессы

Проведенные исследования показали, что предлагаемый модифицированный генетический алгоритм одинаково успешно позволяет оптимизировать настроечные параметры регулирующих устройств как в АСР с традиционными линейными алгоритмами, так и в АСР с нейросетевыми регуляторами. При этом структура регулирующих устройств может быть и более сложной, чем в приведенном примере.

Наряду с применением МГА в задачах анализа и синтеза сложных автоматических систем регулирования, открываются широкие возможности его использования для оптимизации диагностических процедур, реализуемых в АСУТП. В работах [11,12] рассмотрен метод диагностирования информационной подсистемы АСУТП ТЭЦ, в основу которого заложена искусственная нейросеть, обученная на выборке параметрических отказов средств измерения и результатов корректирующих поправок по избыточной информации балансовых соотношений тепломеханического оборудования. По статистическим характеристикам, полученным в последовательных циклах работы алгоритма коррекции, нейросеть формирует вектор признаков состояния каждого информационно-измерительного канала, совокупность которых позволяет проводить процедуры диагностирования. Модифицированный генетический алгоритм позволяет успешно проводить поиск оптимального вектора корректирующих поправок и параметрическую настройку нейросети.

В заключение следует отметить, что для практического использования предложенного алгоритма разработана программа для ЭВМ *Optim-MGA* как инструментальное средство многопараметрической и многоэкстремальной оптимизации. Программа позволяет находить

наилучшее значение функций, представленных в виде динамически присоединяемой библиотеки (dll-файла).

#### Список литературы

- 1. Nelder J.A., Mead R.,** A Simplex Method For Function Minimization, Computer J., No 7, 1964 P. 308-313.
- 2. Сабанин В.Р., Смирнов Н.И.** Расчеты автоматических систем регулирования в теплоэнергетике. МЭИ, 2002.
- 3. Сабанин В.Р., Смирнов Н.И., Репин А.И.** Оптимизация настроечных параметров регулирующих устройств в АСР// Сборник трудов конференции Control 2003. МЭИ, 2003. С. 144-148.
- 4. Goldberg D.E.** Genetic Algorithms in Search Optimizations and Machine Learning.- Addison.Wesly, 1989.
- 5. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / Г.К.Вороновский, К.В. Махотило, С.Н. Петрашев, С.А. Сергеев. – Харьков, Основа, 1997.**
- 6. Махотило К.В.** Разработка методик эволюционного синтеза нейросетевых компонентов систем управления// Диссертация на соискание ученой степени кандидата технических наук.— Харьков, ХГПУ, 1998 .
- 7. Сабанин В.Р., Смирнов Н.И., Репин А.И.** Оптимальный синтез АСР методом имитационного моделирования с использованием генетических алгоритмов оптимизации // Труды всероссийской научной конференции ИММОД 2003. Санкт-Петербург.: ФГУП ЦНИИ технологии судостроения. 2003. С. 87-91.
- 8. Сабанин В.Р., Смирнов Н.И., Репин А.И.** Автоматические системы регулирования на основе нейросетевых технологий // Сборник трудов конференции Control 2003. МЭИ, 2003. С. 45-51.
- 9. Биленко В.А., Давыдов Н.И. и др.** Анализ динамики многосвязной системы регулирования мощности и температуры энергоблока с прямоточным котлом//Теплоэнергетика. 1987. №10. С. 11-17.
- 10. Ротач В.Я., Кузицин В.Ф., Башарин Д.Ю.** Анализ применимости алгоритма автоматизированной настройки для двухсвязной АСР подачи топлива и питательной воды прямоточного котла// Сб. научных трудов “Теория и практика построения и функционирования АСУТП”. МЭИ.1993. С.35-44.
- 11. Сабанин В.Р., Смирнов Н.И., Репин А.И., Аракелян Э.К., Макаров О.Н., Андреев С.Н.** Математическое и программное обеспечение алгоритма коррекции измеряемых параметров для расчета технико-экономических показателей на ТЭЦ// Вестник МЭИ. 2003. №1. С. 21-27.
- 12. Репин А.И., Сабанин В.Р., Смирнов Н.И.** Диагностика информационной подсистемы АСУТП с использованием технологий искусственного интеллекта // Тез. докл. Десятой Международной научно-технической конференции студентов и аспирантов – т3., МЭИ. 2004 с. 145.