

УДК 623:658.562

**В.Г. НАЙДЕНОВ**, доктор  
технических наук, старший  
научный сотрудник  
**К.А. ТАРАСЕНКО**

## **РАЗРАБОТКА АЛГОРИТМА РЕШЕНИЯ ЗАДАЧИ НЕЛИНЕЙНОГО МАТЕМАТИЧЕСКОГО ПРОГРАММИРОВАНИЯ С ОГРАНИЧЕНИЯМИ В ВИДЕ РАВЕНСТВА В ОБЛАСТИ ИЗМЕНЕНИЯ ИССЛЕДУЕМОГО ФУНКЦИОНАЛА**

*В статье разработан алгоритм решения задачи нелинейного математического программирования с ограничениями в виде равенства в области изменения исследуемого функционала. Для решения такой задачи предлагается усовершенствованный алгоритм, реализующий метод Нелдера и Мида. Авторами проведена апробация разработанного алгоритма с использованием математического пакета программ Matlab на примере функции, которая представляет собой длинный узкий «хребет». В результате проведенных исследований разработанного алгоритма можно сделать вывод о том, что такой алгоритм может быть использован при решении задач обоснования требований к техническим характеристикам средств, входящих в состав сложных технических комплексов.*

*Ключевые слова: нелинейное программирование; функционал; деформируемый многоугольник; оптимальное решение.*

В настоящее время существует проблемная задача в выборе подходящего алгоритма для нахождения оптимальных значений искомых величин при решении непрерывных нелинейных задач математического программирования, когда исследуются функционалы, не поддающиеся дифференцированию, а ограничения задаются в виде равенств в области изменения таких функционалов.

Такие проблемы, в частности, возникают при решении так называемых обратных задач и, как правило, связаны с обоснованием требований к техническим характеристикам средств, входящих в состав комплексов, к которым предъявляются исходные требования [1].

Пусть задан нелинейный функционал  $f(x)$ , который не поддается дифференцированию, и вектор  $x$  в пространстве  $E^n$ , являющийся областью определения функционала  $f(x)$ . Требуется найти оптимальное значение вектора  $x_{opt}$ , где значение этого функционала равно требуемой величине  $f_{\text{треб.}}(x)$ . Математически такая задача запишется следующим образом:

$$\text{найти } x_{opt} = \underset{x_{opt} \in \Omega}{\text{Argmax}} f(x) \text{ при ограничении вида } f(x) = f_{\text{треб.}}(x). \quad (1)$$

Задача (1) является специфической задачей нелинейного математического программирования с ограничениями в виде равенства в области изменения исследуемого функционала.

Проведенный анализ литературы [2-4] показал, что в настоящее время не имеется работоспособных алгоритмов, с помощью которых можно решить такую задачу.

В связи с этим возникла идея использовать для решения такой задачи метод Нелдера и Мида, а также реализующий его алгоритм с целью приспособления его для решения задачи нелинейного математического программирования с ограничениями в виде равенств в области изменения исследуемого функционала. Метод Нелдера и Мида свободен от использования дифференцирования исследуемых функционалов и в нем не используются регулярные зондирующие симплексы, которые не позволяют проводить ускорение поиска и проведение поиска на искривленных «оврагах» и «хребтах» исследуемого функционала. В методе Нелдера и Мида симплекс изменяет свою форму в процессе поиска оптимального решения.

В методе Нелдера и Мида, например, при поиске минимума функционала  $f(x)$  значения этого функционала вычисляются в каждой из вершин заданного симплекса. При этом из вершины  $A$ , где функционал имеет максимальное значение, проводится проектируемая прямая через центр «тяжести» симплекса. Затем точка исключается и строится новый симплекс из оставшихся прежних точек и одной новой точки  $B$ , расположенной на проектируемой прямой на определенном расстоянии от центра «тяжести» симплекса. Такой полученный симплекс называется «отражением». Продолжение такой процедуры, в которой каждый раз вычеркивается вершина, где значение функционала максимально,

а также использование правил уменьшения размеров симплекса, регулирования шага и изменения направления поиска оптимального решения, позволяют осуществить поиск, не используя производные от исследуемого функционала. Однако этот метод применим только при решении задач безусловной оптимизации, когда необходимо найти точку в области определения функционала, где имеет место его экстремум.

Не останавливаясь на существующем алгоритме, реализующем метод Нелдера и Мида, который подробно описан в работе [5], перейдем к рассмотрению разработанного алгоритма для сформулированной задачи.

Усовершенствованный алгоритм решения задачи нелинейного математического программирования с ограничением в виде равенства в области изменения исследуемого функционала приведен на рисунке 1.

В разработанном алгоритме находится максимум функционала, зависящего от  $n$  независимых переменных с использованием  $n + 1$  вершин деформируемого многогранника (симплекса) в пространстве  $E^n$ . Каждая вершина может быть идентифицирована вектором  $x$ .

В первой части алгоритма в блоке 1 проводится ввод исходных данных, к числу которых относятся следующие:

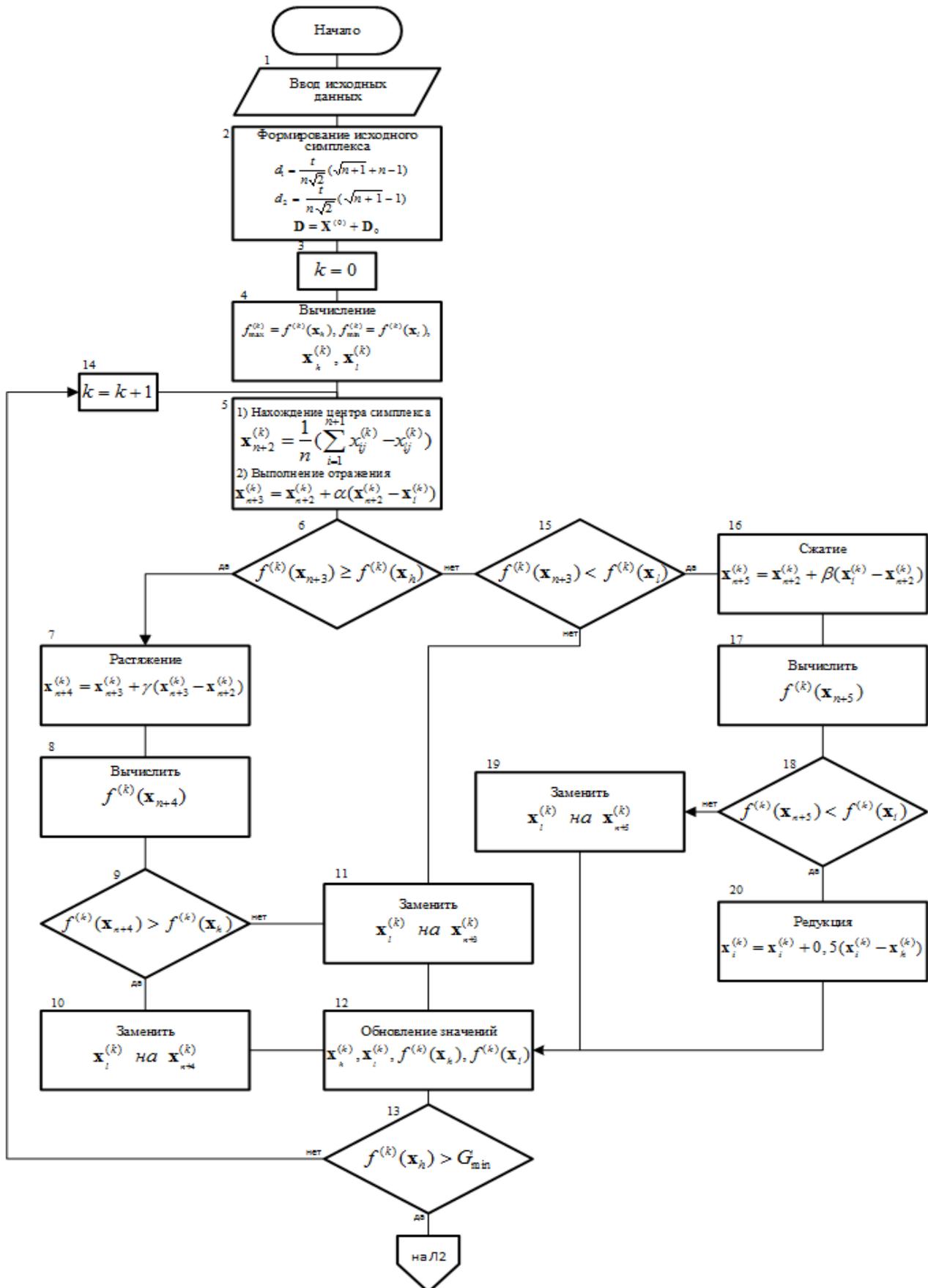
- вид исследуемого функционала и его размерность;
- координаты точки, с которой начинается поиск оптимальных значений в области определения исследуемого функционала;
- значения коэффициентов, регулирующих параметры процесса поиска оптимального решения.

В блоке 2 алгоритма формируется начальный многогранник (зондирующий симплекс)  $D$  размерностью  $n \times (n + 1)$  вида

$$D = X^{(0)} + D_0,$$

где матрицы  $X^{(0)}$  и  $D_0$  имеют следующие структуры:

$$X^{(0)} = \begin{bmatrix} x_{11}^{(0)} & x_{12}^{(0)} & x_{13}^{(0)} & \dots & x_{1(n+1)}^{(0)} \\ x_{21}^{(0)} & x_{22}^{(0)} & x_{23}^{(0)} & \dots & x_{2(n+1)}^{(0)} \\ x_{31}^{(0)} & x_{32}^{(0)} & x_{33}^{(0)} & \dots & x_{3(n+1)}^{(0)} \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1}^{(0)} & x_{n2}^{(0)} & x_{n3}^{(0)} & \dots & x_{n(n+1)}^{(0)} \end{bmatrix}; \quad (2)$$



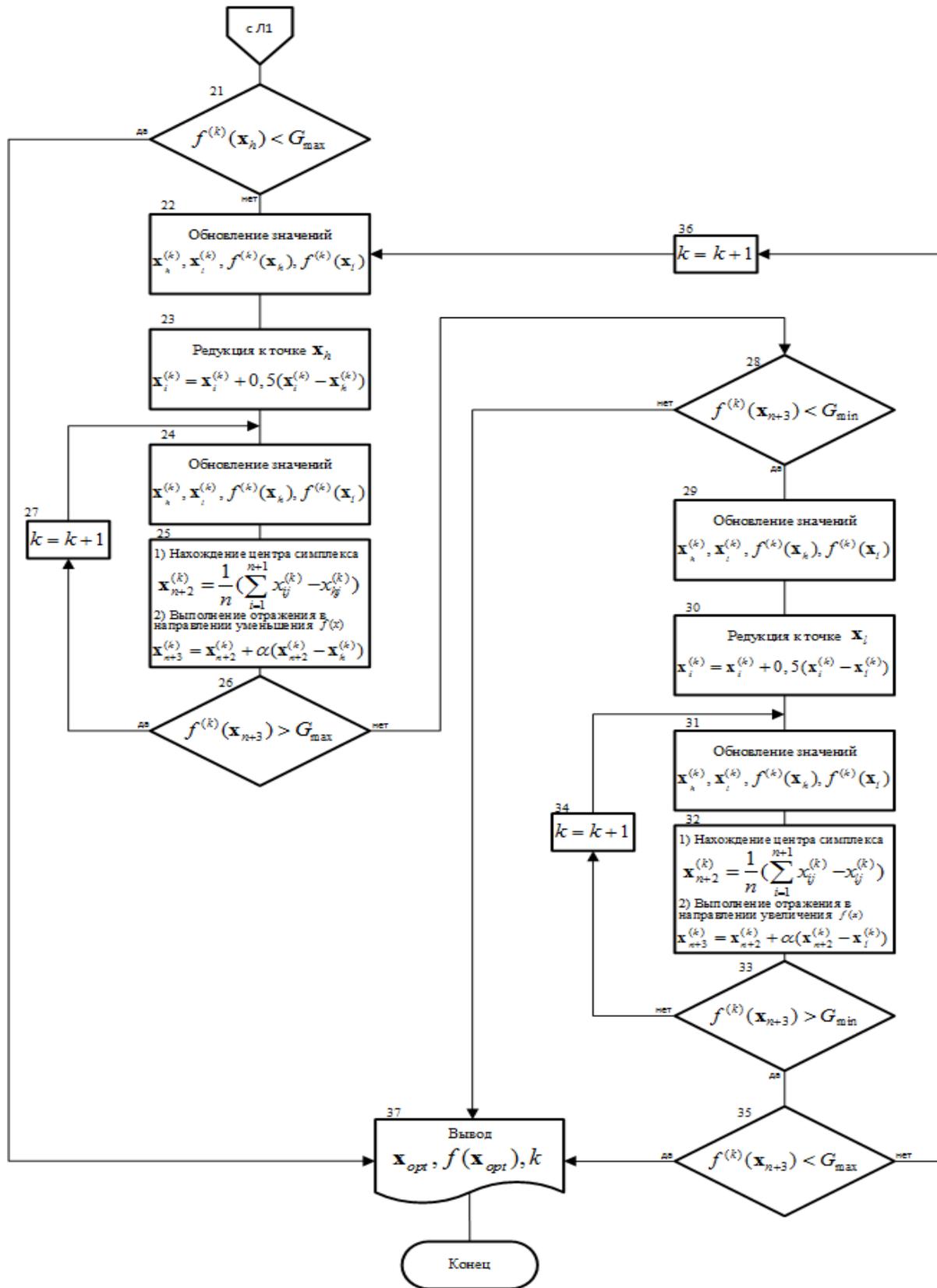


Рисунок 1 – Усовершенствованный алгоритм решения задачи нелинейного математического программирования с ограничением в виде равенства

$$D^{(0)} = \begin{bmatrix} 0 & d_{11}^{(1)} & d_{12}^{(2)} & \dots & d_{1n}^{(2)} \\ 0 & d_{21}^{(2)} & d_{22}^{(1)} & \dots & d_{2n}^{(2)} \\ 0 & d_{31}^{(2)} & d_{32}^{(2)} & \dots & d_{3n}^{(2)} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & d_{n1}^{(2)} & d_{n2}^{(2)} & \dots & d_{nn}^{(1)} \end{bmatrix}. \quad (3)$$

В матрице (2) все столбцы представляют собой начальную точку поиска оптимального решения задачи, а значения  $d_{ij}^{(1)}$  и  $d_{ij}^{(2)}$  в выражении (3) рассчитываются по следующим формулам:

$$d_{ij}^{(1)} = \frac{t}{n\sqrt{2}}(\sqrt{n+1} + n - 1); d_{ij}^{(2)} = \frac{t}{n\sqrt{2}}(\sqrt{n+1} - 1),$$

где  $t$  – расстояние между двумя вершинами симплекса, которое определяется экспериментальным путем.

Пусть  $x_i^{(k)} = [x_{i1}^{(k)}, \dots, x_{ij}^{(k)}, \dots, x_{in}^{(k)}]^T$ , где  $i = 1, \dots, n + 1$ , является  $i$ -й вершиной симплекса в пространстве  $E^n$  на  $k$ -ом этапе поиска, где  $k = 0, 1, \dots$ , и значение целевого функционала в точке  $x_i^{(k)}$  равно  $f(x_i^{(k)})$ . Кроме того, отметим те точки  $x$  симплекса, которые дают максимальное и минимальное значение  $f(x)$ .

В блоке 4 алгоритма проводится определение вершин симплекса  $x_h^{(k)}$  и  $x_l^{(k)}$ , где значения функционала  $f(x)$  имеют максимальное и минимальное значения, исходя из следующего правила:

$$f(x_h^{(k)}) = \max\{f(x_1^{(k)}), \dots, f(x_{n+1}^{(k)})\},$$

$$f(x_l^{(k)}) = \min\{f(x_1^{(k)}), \dots, f(x_{n+1}^{(k)})\}.$$

Затем в блоке 5 алгоритма, исходя из того, что зондирующий многогранник в  $E^n$  состоит из  $(n + 1)$  вершин  $x_1, \dots, x_{n+1}$ , определяется центр «тяжести» всех вершин, исключая точку  $x_l^{(k)}$ , в соответствии с выражением

$$x_{n+2,j}^{(k)} = \frac{1}{n} [(\sum_{i=1}^{n+1} x_{ij}^{(k)}) - x_{li}^{(k)}], j = \overline{1, \dots, n}, \quad (4)$$

где индекс  $j$  означает координатное направление.

Далее в блоке 5 алгоритма выполняется операция «отражения» в направлении увеличения значения исследуемого функционала путем проектирования точки пространства  $x_l^{(k)}$  через центр «тяжести» симплекса в соответствии с соотношением

$$x_{n+3}^{(k)} = x_{n+2}^{(k)} + \alpha(x_{n+2}^{(k)} - x_l^{(k)}),$$

где  $\alpha > 0$  представляет собой коэффициент отражения;  $x_{n+2}^{(k)}$  – центр «тяжести», вычисляемый по формуле (4);  $x_l^{(k)}$  – вершина симплекса, в которой функционал принимает наименьшее значение на  $k$ -ом этапе поиска.

Далее при выполнении условия, что  $f^{(k)}(x_{n+3}) \geq f^{(k)}(x_h)$  (блок 6), выполняется операция «растяжения», когда вектор  $(x_{n+3}^{(k)} - x_{n+2}^{(k)})$  растягивается в соответствии с соотношением

$$x_{n+4}^{(k)} = x_{n+2}^{(k)} + \gamma(x_{n+3}^{(k)} - x_{n+2}^{(k)}),$$

где  $\gamma > 1$  представляет собой коэффициент отражения;

Затем в блоке 8 проводится вычисление значения функционала в точке  $x_{n+4}^{(k)}$ , а в блоке 9 проверяется выполнение условия  $f^{(k)}(x_{n+4}) \geq f^{(k)}(x_h)$ . В случае выполнения этого условия в блоке 10 проводится замена точки  $x_l^{(k)}$  симплекса на точку  $x_{n+4}^{(k)}$ , а в противном случае управление передается на блок 12, где проводится замена точки  $x_l^{(k)}$  симплекса на точку  $x_{n+3}^{(k)}$ . После выполнения операций в блоках 10 или 11 управление передается на блок 12, где проводится обновление значений  $x_h^{(k)}, x_l^{(k)}, f^{(k)}(x_h), f^{(k)}(x_l)$ .

Далее в блоке 13 проводится сравнение значение функционала в точке  $x_h^{(k)}$  со значением минимального уровня  $G_{\min}$  границы ограничения значений изменения функционала. При невыполнении условия блока 13 организуется выполнение следующего этапа цикла движения в сторону увеличения значения исследуемого функционала.

В случае невыполнения условия блока 6 проводится проверка выполнения условия  $f^{(k)}(x_{n+3}) < f^{(k)}(x_l)$  (блок 15). Если последнее условие выполняется, то выполняется операция «сжатия» зондирующего симплекса в соответствии с выражением

$$x_{n+5}^{(k)} = x_{n+2}^{(k)} + \beta(x_h^{(k)} - x_{n+2}^{(k)}),$$

где  $0 < \beta < 1$  – представляет собой коэффициент сжатия.

Далее в блоке 17 вычисляется значение функционала в точке  $x_{n+5}^{(k)}$ , а в блоке 17 осуществляется проверка выполнения условия  $f^{(k)}(x_{n+5}) < f^{(k)}(x_l)$ . В случае выполнения этого условия проводится операция редукции (сжатия симплекса) в соответствии с выражением

$$x_i^{(k)} = x_i^{(k)} + 0,5 \cdot (x_i^{(k)} - x_h^{(k)}), i = \overline{1, \dots, n+1},$$

а в противном случае проводится замена точки  $x_l^{(k)}$  симплекса на точку  $x_{n+5}^{(k)}$ . Затем управление с двух последних блоков передается на блок 12 алгоритма.

Во второй части алгоритма обеспечивается поиск максимума функционала в области заданного ограничения, которая задается в виде минимальной границы  $G_{\min}$  и максимальной границы  $G_{\max}$ . Разность между значениями  $G_{\max}$  и  $G_{\min}$  определяет величину  $\varepsilon$ , которая характеризует допустимую точность принятия решения об обнаружении оптимального значения исследуемого функционала.

Так, в случае выполнения условия  $f^{(k)}(x_h) < G_{\max}$  (блок 21) проводится вывод оптимального значения функционала  $f(x_{\text{opt}})$ , координаты точки  $x_{\text{opt}}^{(k)}$ , а также числа  $k$  отработанных алгоритмом циклов вычислений (блок 37).

В случае невыполнения условия блока 21 проводится обновление значений  $x_h^{(k)}, x_l^{(k)}, f^{(k)}(x_h), f^{(k)}(x_l)$ , а затем осуществляется выполнение операции редукции в точке  $x_h^{(k)}$  (блок 23) и после этого проводится повторное обновление значений  $x_h^{(k)}, x_l^{(k)}, f(x_h), f(x_l)$  в блоке 24.

Далее в блоке 25 алгоритма проводится операция нахождения центра нового симплекса в соответствии с приведенным выше выражением, а также выполнение операции «отражения» в направлении уменьшения значения исследуемого функционала. Затем в случае выполнения условия  $f^{(k)}(x_{n+3}) > G_{\max}$  осуществляет циклическое продолжение поиска оптимальной точки в области изменения функционала в направлении уменьшения его значения, а в случае невыполнения этого условия проводится сравнение значения функционала в точке  $x_{n+3}^{(k)}$  с минимальной границей  $G_{\min}$  области ограничений (блок 28).

При невыполнении условия блока 28 управление передается на блок 37, который обеспечивает выдачу искомым данных  $f(x_{\text{opt}}), x_{\text{opt}}$  и  $k$ , а в противном случае проводится повторное обновление значений  $x_h^{(k)}, x_l^{(k)}, f^{(k)}(x_h), f^{(k)}(x_l)$  (блок 29).

Далее в блоке 30 проводится выполнение операции редукции в точке  $x_l^{(k)}$  (блок 30) и повторное обновление значений  $x_h^{(k)}, x_l^{(k)}, f^{(k)}(x_h), f^{(k)}(x_l)$  в блоке 31.

В блоке 32 алгоритма проводится операция нахождения центра нового симплекса, а также выполнение операции «отражения» в направлении увеличения значения исследуемого функционала. Затем в случае невыполнения условия  $f^{(k)}(x_{n+3}) > G_{\min}$  (блок 33) осуществляет циклическое продолжение поиска оптимальной точки в направлении увеличения значения исследуемого функционала, а в случае выполнения этого условия проводится сравнение значения функционала в точке  $x_{n+3}^{(k)}$  с минимальной границей  $G_{\max}$  области ограничений (блок 35). Если условие блока 35 не выполняется, то организуется дальнейший циклический процесс поиска оптимального значения функционала в области наложенных ограничений. При выполнении условия блока 35 проводится вывод оптимального значения функционала  $f(x_{\text{opt}})$  координаты точки  $x_{\text{opt}}$ , а также числа  $k$  отработанных алгоритмом циклов вычислений. На этом функционирование алгоритма заканчивается.

Апробация разработанного алгоритма была проведена с использованием математического пакета программ Matlab на примере функции вида

$$f(x) = x_1^2 \exp[1 - x_1^2 - 20,25(x_1 - x_2)^2],$$

которая представляет собой длинный узкий «хребет» с максимумом в точке  $x^* = [11]^T$ .

Требуется найти точку в области определения функции  $f(x)$ , где ее значение принимает максимальное значение при ограничении  $f_{\text{огр}}(x) = 0,8$ .

При решении данной задачи были приняты следующие параметры процесса поиска оптимального решения:  $t = 0,2; \alpha = 1,0; \beta = 0,5; \gamma = 2,0$ . Величина зоны ограничения  $\varepsilon$  между верхней  $G_{\max}$  и нижней  $G_{\min}$  границами, которая определяет точность решения задачи, равна 0,0005.

На рисунке 2 показан внешний вид исследуемой функции и процесс поиска оптимального решения при заданном ограничении на значение функции.

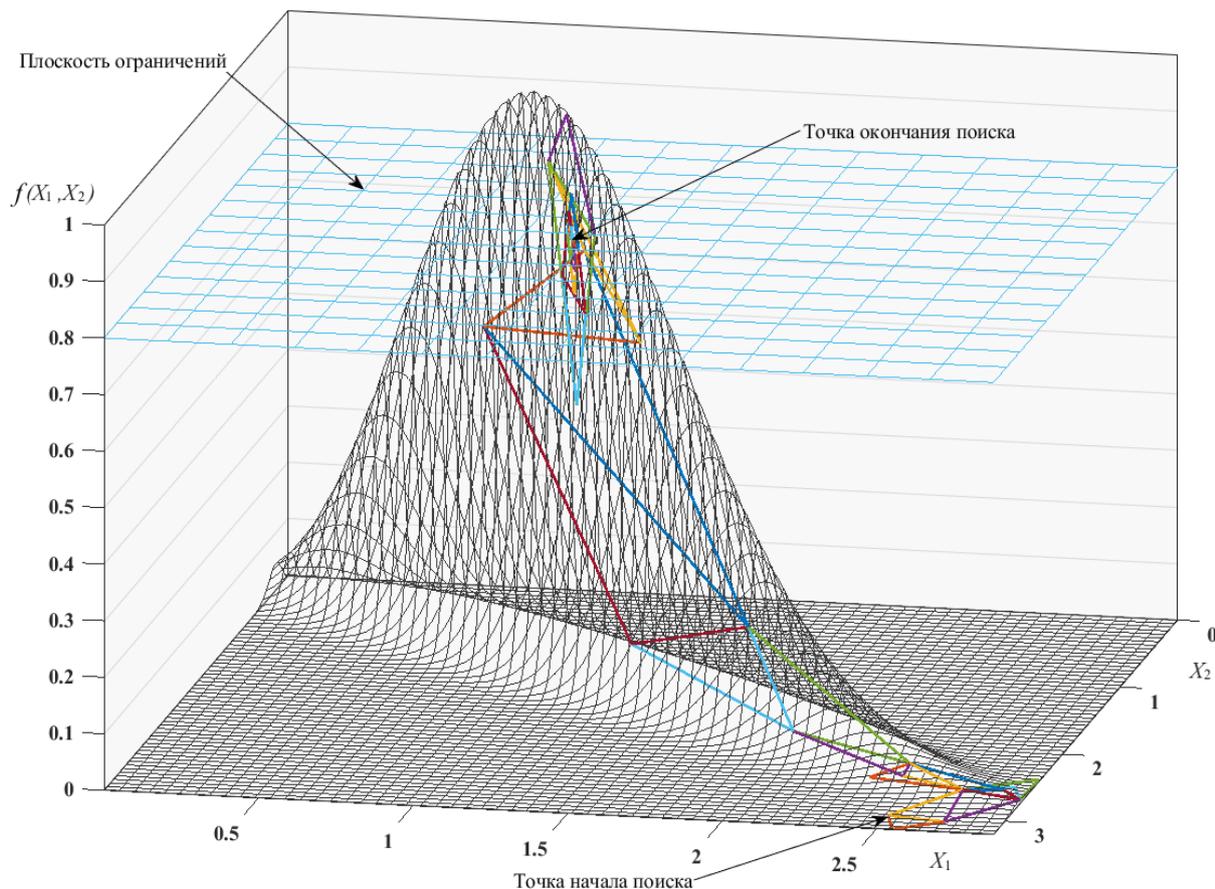


Рисунок 2 – Внешний вид исследуемой функции и визуализация процесса поиска оптимального решения

Исходной точкой, откуда начинался процесс поиска решения, являлась точка  $x_{исх.} = [2,53]^T$ . В результате поиска, содержащего 36 шагов, было получено оптимальное решение  $x^* = [1,2291,151]^T$ .

На рисунках 3 и 4 показана визуализация процесса поиска оптимального решения задачи для двух различных исходных точек поиска, равных  $x_{исх.} = [32,5]^T$  и  $x_{исх.} = [2,53]^T$ , а в таблице 1 показаны параметры и результаты решения задачи нахождения оптимального решения задачи.

Таблица 1 – Параметры и результаты решения задачи нахождения оптимального решения задачи

Точка начала поиска и размер исходного симплекса			Точка окончания поиска, значение функции в этой точке и количество шагов поиска			
$x_{1(исх.)}$	$x_{2(исх.)}$	$t$	$x_{1(opt)}$	$x_{2(opt)}$	$f_{opt}(x_1, x_2)$	$k$
3	2.5	0.2	1.2170	1.1352	0.7996	40
2.5	3	0.2	1.2294	1.1509	0.8001	36

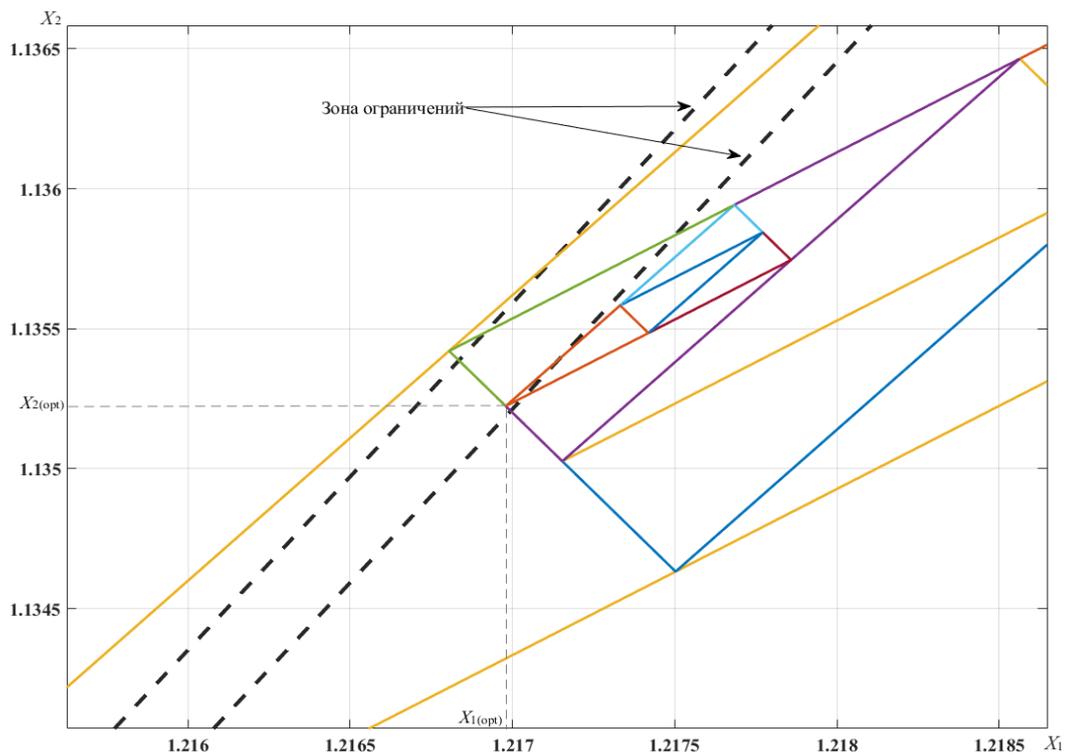


Рисунок 3 – Зона ограничений исследуемой функции и визуализация процесса поиска оптимального решения задачи при исходной точке поиска, равной  $x_{исх.} = [32,5]^T$

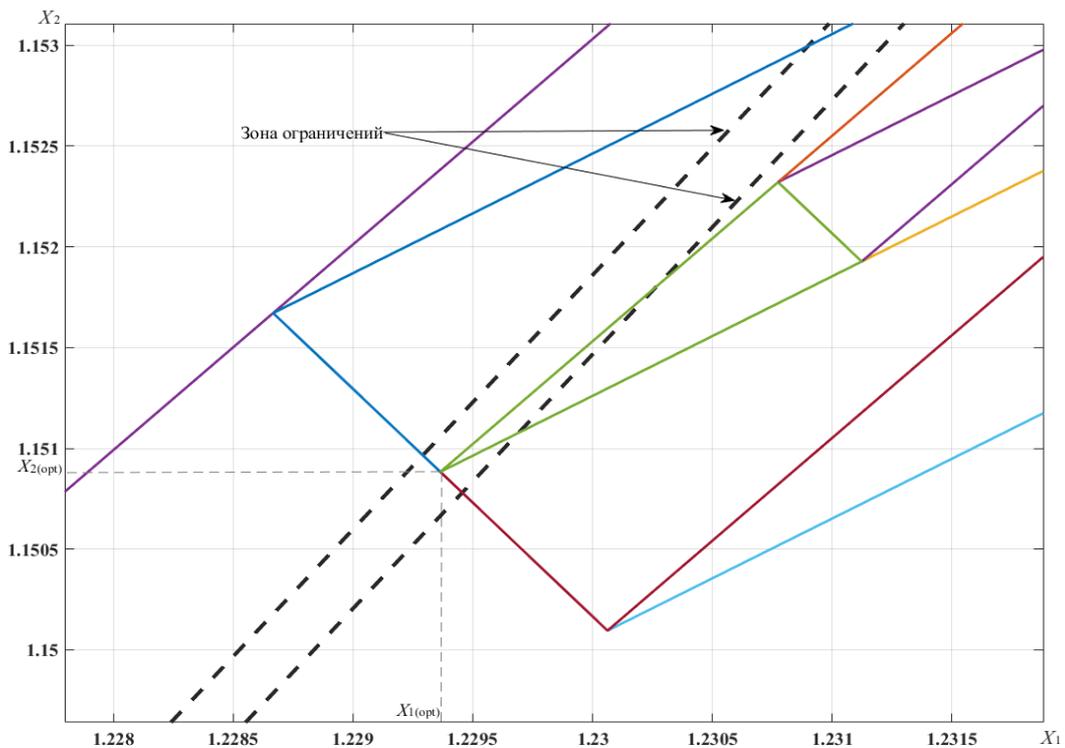


Рисунок 4 – Зона ограничений исследуемой функции и визуализация процесса поиска оптимального решения задачи при исходной точке поиска, равной  $x_{исх.} = [2,53]^T$

Результаты апробации разработанного алгоритма показывают его работоспособность, высокую точность и повторяемость получаемых решений.

Таким образом, в результате проведенных исследований разработанного алгоритма решения задачи нелинейного математического программирования с ограничениями в виде равенства в области изменения исследуемого функционала можно сделать вывод о том, что такой алгоритм может быть с успехом использован при решении задач обоснования требований к техническим характеристикам средств, входящих в состав сложных технических комплексов, к которым уже предъявлены исходные требования.

#### **Список использованных источников**

1. Найденов В.Г., Тарасенко К.А., Першин Е.В. Методический подход к обоснованию требований к перспективным наземным средствам траекторных измерений по точности получения ими первичных измерений // Вооружение и экономика. 2021. №1.
2. Таха Х. Введение в исследование операций. Кн. 1, 2. М.: Мир, 1975.
3. Вентцель Е.С. Исследование операций. М.: Советское радио, 1972.
4. Эльстер К.Х. и др. Введение в нелинейное программирование. М.: Наука, 1985.
5. Химмельблау Д. Прикладное нелинейное программирование / Под ред. М.Л. Быховского. М.: Мир, 1975.