

УДК 519.857

## ИСПОЛЬЗОВАНИЕ ПАРАБОЛИЧЕСКИХ СПЛАЙНОВ В САПР ЛИНЕЙНЫХ СООРУЖЕНИЙ

**В.И. Струченков**

*Московский технологический университет (МИРЭА), Москва 119454, Россия  
@Автор для переписки, e-mail: str1942@mail.ru*

В статье рассматриваются математические модели и алгоритмы оптимизации, используемые при проектировании продольного профиля новых и реконструируемых автомобильных дорог и других линейных сооружений транспорта. В качестве базового представления искомой проектной линии продольного профиля проектируемой трассы принят параболический сплайн, то есть плоская кривая, элементами которой являются параболы второй степени. Параметры этих элементов должны удовлетворять техническим ограничениям, которые формализуются в виде системы линейных неравенств. Искомый сплайн является экстремалью некоторого функционала, который получается при формализации таких технико-экономических критериев оптимальности, как объемы земляных работ и строительные затраты. Принципиальной особенностью рассматриваемой задачи оптимизации является отсутствие сведений о числе элементов искомого сплайна. В этой связи предлагается многоэтапный подход: сначала определить число элементов, а затем оптимизировать их параметры. Число элементов определяется по результатам работы программ, проектирующих ломаную линию, состоящую из элементов малой длины, с соблюдением всех технических ограничений. Число этапов оптимизации параметров при известном числе элементов сплайна зависит от конкретной задачи, принятого критерия оптимальности, а также полноты и детальности исходных данных. Рассматриваются алгоритмы динамического и нелинейного программирования, особенности их реализации и результаты применения разработанных компьютерных программ.

**Ключевые слова:** трасса, продольный профиль, сплайн, динамическое и нелинейное программирование, целевая функция, градиент.

## THE USE OF PARABOLIC SPLINEES IN CAD OF LINEAR STRUCTURES

**V.I. Struchenkov**

*Moscow Technological University (MIREA), Moscow 119454, Russia  
@Corresponding author e-mail: str1942@mail.ru*

The article continues studying the problem of the calculus of variations that occurs in line structures routing, in particular, roads. The task is to find a line that satisfies all technical constraints and gives a minimum of a given functional, for example, construction costs. The unknown extremal is a parabolic spline, that is, a plane curve, the elements of which are parabolas of the second order conjugated by line segments. The principal feature of the problem is that the number of spline elements is unknown. The spline parameters must satisfy the constraints on the first derivative and curvature. Besides, also the ordinates of the individual points may be restricted. In addition, the lengths of the spline elements must be at least the given values. The problem is solved in two stages. First, the number of elements is determined, and then their parameters are optimized. Algorithms of nonlinear and dynamic programming are used. The structural features of the constraint system are taken into account, and an algorithm for constructing a basis in the null space of the matrix of active constraints is given. As an alternative, an algorithm is implemented that uses penalty functions for violation of constraints on ordinates of given points. The successful implementation of algorithms is reported.

**Keywords:** route, horizontal and vertical alignment, nonlinear programming, objective function, gradient, feedbacks.

### Введение

Настоящая статья является логическим продолжением ранее опубликованной работы<sup>1</sup>. При автоматизированном проектировании трасс таких линейных сооружений, как автомобильные дороги различного назначения, траншеи для магистральных трубопроводов, трасса сооружения как трехмерная кривая традиционно представляется двумя плоскими кривыми: планом и продольным профилем [1–3]. Варианты положения трассы в плане назначаются специалистами, исходя из технико-экономических, экологических и других условий. В обжитых районах, как правило, этих вариантов немного. Достаточно вспомнить историю с Химкинским лесом, когда альтернативный вариант так и не был найден. Для выбора варианта из нескольких возможных по каждому из них проектируется продольный профиль, то есть плоская кривая, являющаяся однозначной функцией координаты  $Z$  от длины дуги в плане  $s$ , начиная от заданной точки. Кривая состоит из нескольких элементов, в точке сопряжения элементы имеют общую касательную. В качестве элементов могут быть дуги окружностей, отрезки прямых, парабол, клотоид и др.

В настоящей статье в качестве элементов искомой проектной линии приняты параболы второй степени и отрезки прямых, как частный случай парабол. Параметры элементов должны удовлетворять условиям гладкости, то есть ограничениям на первую производную и кривизну. Кроме того, могут быть ограничения на координату  $Z$  в отдельных точках. Длины элементов не могут быть меньше заданных величин, при этом число элементов неизвестно.

Цель статьи – показать, как определить число элементов с помощью метода динамического программирования, получить начальное приближение искомой линии в виде параболического сплайна и оптимизировать параметры сплайна, используя методы нелинейного программирования. Это соответствует методологии разработки проектирующих

<sup>1</sup>Струченков В.И. Компьютерные технологии проектирования трасс линейных сооружений // Российский технологический журнал. 2017. Т. 5. № 1. С. 29–41.

систем [4], которые принципиально отличаются от действующих САПР [5–8], где проектные решения тем или иным способом назначаются проектантом, причем компьютер используется для решения вспомогательных задач, а не для выработки проектных решений на основе математических моделей и алгоритмов оптимизации.

### 1. Формализация задачи

Обозначим профиль земли по заданному варианту положения трассы в плане как  $Z(s)$ , а искомую проектную линию –  $P(s)$ . Задача состоит в следующем: по заданной  $Z(s)$  найти такую  $P(s)$ , которая удовлетворяет всем ограничениям и дает

$$\int_0^{S_0} F(Z(s), P(s), s) ds \tag{1}$$

где  $S_0$  – длина трассы в плане;

функция  $F$  моделирует строительные затраты на элементе длины;

$Z(s)$  – ломаная линия с неравномерным шагом;

$P(s)$  – последовательность парабол второй степени, сопрягаемых встык (с общей касательной) или отрезками прямых, которые рассматриваются как параболы нулевой кривизны.

Задача вариационного исчисления (1) в конечном итоге будет сведена к задаче нелинейного программирования, обладающей интересными особенностями независимо от конкретного вида функции  $F$ . Прежде всего, надо определить размерность задачи, то есть число элементов искомого сплайна. Первоначально вместо параболического сплайна будем искать ломаную линию, у которой абсциссы переломов совпадают с абсциссами переломов профиля земли. При необходимости количество элементов этой ломаной можно увеличить для более точной аппроксимации искомого сплайна.

Совпадение абсцисс профиля земли и искомой ломаной (замены сплайна) позволяет задать число элементов  $n$  (размерность задачи) и их длины  $s_i$  (в плане). При этом получится не сплайн, а его приближение в виде ломаной линии из коротких элементов. Такая линия называется «цепочечный профиль» [9]. Число звеньев ломаной существенно больше, чем число элементов сплайна, но при соблюдении ограничений, которые являются дискретными аналогами ограничений по первой производной (ограничения на уклоны элементов ломаной) и кривизну (ограничения на разность уклонов смежных элементов ломаной) отклонения от сплайна  $P(s)$  невелики [9]. Идея заключается в том, чтобы найти ломаную путем решения задачи оптимизации, затем преобразовать ее в сплайн, определив тем самым реальную размерность задачи и начальное приближение, и на последнем этапе выполнить оптимизацию параметров сплайна при всех ограничениях и необходимых уточнениях целевой функции.

#### 1.1. Система ограничений

В качестве неизвестных  $p_i$  ( $i = 1, 2, \dots, n$ ) принимаем ординаты искомой ломаной в точках перелома. Ограничения делятся на три группы:

- 1) на ординаты в отдельных точках  $p_i \leq p_i^{\max}$  или  $p_i \geq p_i^{\min}$ ;

2) на уклоны элементов профиля (являются дискретным аналогом ограничения на первую производную)

$$a_i \leq (p_{i+1} - p_i)/s_i \leq b_i \quad (i = 1, 2, \dots, n-1),$$

здесь  $s_i$  – длины элементов;

3) на разности уклонов смежных элементов:

$$c_i \leq (p_{i+2} - p_{i+1})/s_{i+1} - (p_{i+1} - p_i)/s_i \leq d_i.$$

Эти ограничения являются дискретным аналогом ограничения на кривизну.

В силу малости проектных уклонов длина элемента и длина его проекции на ось абсцисс практически совпадают.

## 1.2. Критерий оптимальности

Переход от сплайна к ломаной из коротких элементов позволяет свести вариационную задачу (1) к задаче нелинейного программирования с линейной системой ограничений. Интеграл (1) преобразуется в целевую функцию  $K(\mathbf{h})$  и задача приобретает вид:

$$K(\mathbf{h}) = \sum_{i=1}^{n-1} f_i(h_i, h_{i+1}, s_i) + \sum_{j=1}^{j=k} g_j(h_j) \quad (2)$$

при  $\mathbf{A}\mathbf{h} \leq \mathbf{b}$ .

Здесь вектор неизвестных  $\mathbf{h} (h_1, h_2, \dots, h_n)$  имеет компоненты  $h_i = p_i - z_i$  – разности ординат искомого профиля («цепочки») и профиля земли в точках его перелома. Переменные  $h_i$  как отклонение проектной линии от профиля земли называются рабочими отметками. Функции  $f_i(h_i, h_{i+1}, s_i)$  моделируют затраты на сооружение земляного полотна, а функции  $g_j(h_j)$  моделируют затраты на искусственные сооружения (водопрпускные трубы, мосты и др.), число которых ( $k$ ), типы и местоположение считаются заданными. В последней сумме участвуют только те переменные, которые соответствуют местоположению искусственных сооружений.

Поскольку  $z_i$  известны, система линейных ограничений на компоненты вектора  $\mathbf{p}$  преобразуется в систему линейных ограничений на компоненты вектора  $\mathbf{h}$  с заданными матрицей  $\mathbf{A}$  и вектором  $\mathbf{b}$ . Система линейных ограничений имеет простую структуру, что позволяет модифицировать известный алгоритм приведенного градиента [10, 11]. В отличие от стандартных алгоритмов решения задач данного типа с произвольными системами линейных ограничений [5] здесь реализован алгоритм, не требующий решения систем линейных уравнений для поиска направления спуска на каждой итерации. Направление спуска (приведенный антиградиент) вычисляется по простым формулам, что позволяет решать задачу нелинейного программирования с числом переменных до 1000 и числом ограничений до 4500. Это соответствует проектированию 15-20 км автодорог за несколько секунд на общедоступных персональных компьютерах.

## 2. Расчет параболического сплайна

Будем считать заданными начальную точку и направление касательной в ней (точка А и уклон  $i_1$  на рис. 1), а также конечную точку В. Может быть задано и направление касательной в точке В. Задача состоит в построении параболического сплайна, который удовлетворяет всем ограничениям, включая ограничения на минимальную длину элемента ( $l_{min}$ ), и наилучшим образом аппроксимирует «цепочечный профиль» (кривая АВ на рис.1), полученный на предыдущем этапе расчетов.

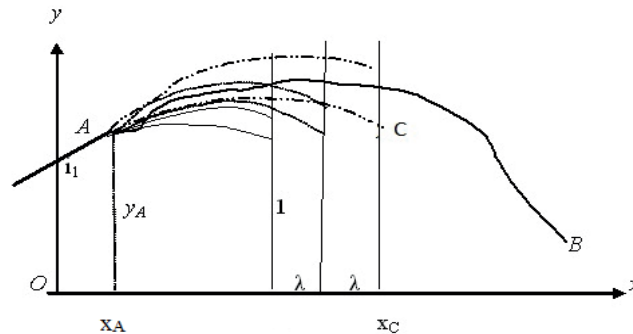


Рис. 1. Варианты первой параболы.

К сожалению, в общем случае на данном этапе в качестве критерия качества аппроксимации не удастся использовать целевую функцию (строительные затраты), которая использовалась при проектировании «цепочечного» профиля. Дело в том, что при сооружении земляного полотна грунт выемок обычно используется для сооружения насыпей, а если грунта выемок, пригодного для сооружения насыпей, недостаточно, то используются карьеры (резервы). Следовательно, для вычисления затрат на сооружение земляного полотна нужно знать соотношение объемов насыпей и выемок, то есть проектную линию в целом. Мы же будем строить параболический сплайн поэлементно, используя динамическое программирование. При заданных конструкциях поперечных профилей земляного полотна в качестве целевой функции можно принять суммарный объем насыпей и выемок, однако в этом случае расчеты существенно усложняются. Следовательно, в качестве критерия близости искомого параболического сплайна и «цепочечного» профиля целесообразно принять сумму квадратов разностей ординат с заданным постоянным шагом. Иными словами, относительно аппроксимируемой линии разобьем регулярную сетку с заданным шагом по оси абсцисс ( $\lambda$  на рис. 1) и ординат  $\Delta$ .

Первая вертикаль сетки отстоит от точки А по оси абсцисс на  $l_{min}$ , далее с шагом  $\lambda$ . На первом шаге (рис. 1) в каждый узел сетки на вертикали 1 приходит только одна парабола, так как начальная точка А и начальное направление заданы. Уравнение параболы  $y = ax^2 + bx + c$ . Если задать ординату конца параболы  $y_C$ , то для определения параметров  $a$ ,  $b$  и  $c$  имеем систему из трех уравнений:

$$ax_A^2 + bx_A + c = y_A;$$

$$ax_C^2 + bx_C + c = y_C;$$

$$2ax_A + b = i_{нач};$$

здесь начальный уклон  $i_{нач} = i_1 = \operatorname{tg}\alpha$ ,

$\alpha$  – угол заданного начального направления с осью  $x$ .

Находим последовательно

$$a = \frac{y_C - y_A - i_{нач}(x_C - x_A)}{(x_C - x_A)^2}$$

$$b = i_{нач} - 2ax_A \text{ и } c = y_A - x_A^2 - bx_A.$$

Уклон в конце элемента  $i_{кон} = 2(y_C - y_A)/(x_C - x_A) - i_{нач}$ .

Если  $(y_C - y_A)/(x_C - x_A) = i_{нач}$ , то вместо параболы имеем прямую ( $i_{кон} = i_{нач}$ ) как частный случай. Отметим, что при изменении  $y_C$  на  $\Delta$  (шаг сетки) конечный уклон меняется на  $2\Delta/(x_C - x_A)$ . Естественно, что из всех парабол остаются только те, параметры которых удовлетворяют всем ограничениям, и для каждой допустимой параболы вычисляется целевая функция. Ограничения на кривизну фактически сводятся к двусторонним ограничениям на параметр  $a$ , если уклоны много меньше единицы, что имеет место в задачах проектирования трасс линейных сооружений. В любом случае, для любого построенного варианта параболы легко проверить выполнение всех заданных ограничений.

Кроме вертикали 1, первый элемент может заканчиваться на всех последующих вертикалях. Однако нет смысла строить элементы длиной больше чем  $2l_{min}$ , так как каждый из них может быть получен как два элемента. Следовательно, на каждой вертикали в интервале  $x_A + l_{min} \leq x \leq x_A + 2l_{min}$  при построении первого элемента в каждый узел сетки приходит одна парабола, которая запоминается, если не нарушены ограничения.

При построении второго и всех последующих элементов каждая точка на вертикали 1 должна рассматриваться как начальная. На первом элементе это была точка А с одним возможным значением начального уклона, теперь таких точек много, и в каждой из них есть несколько значений начального уклона. Если число точек на каждой вертикали равно  $m$ , то без учета ограничений на второй вертикали в **каждой точке** есть  $m$  сходящихся элементов (каждый со своим конечным уклоном), на третьей соответственно  $m^2$ , потом  $m^3$  и так далее.

В динамическом программировании ключевым понятием является «состояние системы». В рассматриваемой задаче в качестве «состояния системы» нельзя принять отдельную точку на вертикали, нужно еще и значение уклона (производной) в ней. Сравнить можно только варианты, сходящиеся в точке с одним и тем же уклоном, и из всех таких вариантов – оставлять для дальнейшего рассмотрения только один, но при таком правиле отбраковки резкое возрастание количества вариантов приводит к непреодолимым вычислительным трудностям.

Исходя из выше изложенного, будем считать сравнимыми сходящиеся в одной точке варианты, если их конечные уклоны близки, т.е. введем дискретность по уклонам. Следовательно, задача становится двухпараметрической, хотя по второму параметру (уклон) сетка варьирования строится в процессе счета. Величина дискрета по уклонам должна выбираться так, чтобы вызванное этой дискретностью максимальное отклонение на следующем элементе не превышало шага сетки  $\Delta$ .



На каждой вертикали в каждом узле сетки запоминается «веер» входящих парабол, их конечные уклоны, номер вертикали начала элемента и номер узла на ней плюс сумма квадратов отклонений от аппроксимируемой кривой по всем пройденным вертикалям (значение целевой функции).

Процесс продолжается до вертикали, отстоящей от конечной точки на  $L_{min}$  (рис. 2), но начиная с  $2L_{min}$  новый элемент строится с условием выхода в конечную точку и, если требуется, то и в конечном направлении ( $i_{fin}$ ). После сравнения всех допустимых вариантов последнего элемента и выбора наилучшего из них обратным разворотом определяем все элементы, а именно наилучший в заданных условиях параболический сплайн.

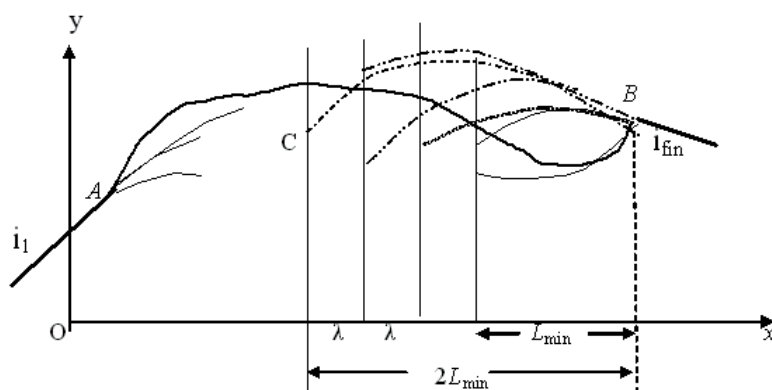


Рис. 2. Построение последнего элемента.

### 3. Оптимизация параметров сплайна

Будем рассматривать полученный сплайн как начальное приближение задачи нелинейного программирования, к которой сводится исходная вариационная задача. Возможность оптимизации длин элементов сплайна пока рассматривать не будем. В качестве переменных примем уклоны (первые производные) в начале каждого элемента плюс уклон в конце последнего элемента. Если начальный или конечный уклоны (один или оба) фиксируются, то дополнительных сложностей не возникает, просто уменьшается размерность задачи.

Уравнение каждой параболы в декартовой системе координат ( $l, H$ ), помещенной в начало элемента, имеет вид:

$$H = al^2 + bl + c,$$

где  $c$  – проектная отметка в начальной точке элемента;

$b$  – уклон в начале элемента;

$l$  – расстояние от начала элемента.

Если считать неизвестными все уклоны – параметры  $b_j$  ( $j = 1, 2, \dots, N+1$ ), где  $N$  – число элементов сплайна, то через них можно найти проектную отметку любой точки, считая заданной отметку начальной точки. Действительно, параметр  $a_j$  каждой параболы вычисляется через уклоны в ее начальной и конечной точках, так как длина элемента  $L_j$  известна:

$$a_j = (b_{j+1} - b_j) / (2L_j) \tag{3}$$

$$c_{j+1} = (b_{j+1} + b_j)L_j/2 + c_j \quad (4)$$

При любой комбинации параметров  $b_j$  можно вычислить все рабочие отметки, которые раньше считались независимыми переменными, а, следовательно, и частные производные (градиент) целевой функции. Нужно только последовательно вычислять параметры  $a_j$  и  $c_{j+1}$  по формулам (3) и (4).

Проектная отметка  $P_Q$  произвольной точки Q определяется по формуле

$$P_Q = P_0 + \sum_{j=1}^{r-1} \frac{(b_{j+1} + b_j)L_j}{2} + b_r S_Q + \frac{b_{r+1} - b_r}{2L_r} S_Q^2 \quad (5)$$

здесь  $P_0$  – проектная отметка в начале профиля;

$r$  – номер элемента, содержащего точку Q;

$S_Q$  – расстояние точки Q от начала этого элемента.

При  $r = 1$  имеем просто

$$P_Q = P_0 + b_1 S_Q + (b_2 - b_1) S_Q^2 / (2L_1) \quad (6)$$

Ограничения по уклонам записываются как

$$b_j^{\min} \leq b_j \leq b_j^{\max} \quad (7)$$

Ограничения по радиусам

$$L_j / R_j^{\min 1} \leq b_{j+1} - b_j \leq L_j / R_j^{\min 2}. \quad (8)$$

здесь  $R_j^{\min 1} < 0$  и  $R_j^{\min 2} > 0$  – минимально допустимый радиус выпуклой и вогнутой кривой, соответственно. Их можно задать различными на различных участках профиля. В частности, задав  $R_j^{\min 1} > R_j^{\min 2} > 0$ , запрещаем выпуклую кривую на соответствующем элементе, а задав  $R_j^{\min 2} < R_j^{\min 1} < 0$ , запрещаем вогнутую кривую.

Формулы (5) и (6) используются для пересчета частных производных, а также для проверки активности высотных ограничений. Из них следует, что при известных длинах элементов  $L_j$  высотные ограничения выражаются линейными неравенствами относительно неизвестных  $b_j$ . Каждое из таких ограничений связывает уклоны всех предшествующих элементов. Например, условие выхода в конечную точку связывает уклоны всех элементов, что является существенным осложнением по сравнению с крайне простой структурой системы ограничений (7), (8).

Широко известные алгоритмы нелинейного программирования с линейной системой ограничений [11, 12] представляют собой итерационный процесс, состоящий из следующих шагов:

- 1) вычисление допустимого начального приближения  $\mathbf{z}^0$ ;
- 2) вычисление градиента целевой функции  $\mathbf{f}^0$ ;
- 3) определение множества активных ограничений (активного набора);
- 4) построение направления спуска  $\mathbf{p}^0$  в граничном линейном многообразии;



5) проверка условий прекращения счета и, если они выполнены, то окончание процесса, иначе – следующий шаг;

6) вычисление шага  $s$  и новой итерационной точки  $\mathbf{z}^{k+1} = \mathbf{z}^k + s\mathbf{p}^k$  и переход к пункту 2.

Эти алгоритмы отличаются способом построения вектора спуска. Если в качестве направления спуска используется проекция  $\mathbf{p}$  антиградиента  $-\mathbf{f}$ , то стандартный алгоритм для вычисления проекции требует на каждой итерации решать систему линейных уравнений. Однако, если на каждой итерации номер  $k$  для соответствующего набора активных ограничений, составляющих матрицу  $\mathbf{A}_k$ , мы умеем строить базис в ее нуль-пространстве, то в качестве направления спуска можно использовать приведенный антиградиент [9] и обойтись без решения систем уравнений. Поскольку направление спуска  $\mathbf{p}^k$  удовлетворяет всем активным ограничениям как равенствам, то  $\mathbf{A}\mathbf{p}^k = 0$ . Если  $\mathbf{C}$  – базисная матрица нуль-пространства матрицы  $\mathbf{A}_k$ , то приведенный антиградиент вычисляется по формуле  $\mathbf{p}^k = \mathbf{C}\mathbf{C}^T\mathbf{f}^k$ , где  $\mathbf{f}^k$  – антиградиент целевой функции на  $k$ -ой итерации, а  $\mathbf{C}^T$  – транспонированная базисная матрица.

При временном исключении из рассмотрения ограничений на ординаты отдельных точек (высотные ограничения) и требования выхода в конечную точку оставшаяся система ограничений чрезвычайно проста. В таком случае нет необходимости строить базисную матрицу, так как можно на каждой итерации просто вычислить проекцию антиградиента на нуль-пространство матрицы активных ограничений при любой их комбинации. Характерно, что в этом случае при наличии переменной, которая не входит ни в одно из активных ограничений, компоненты проекции антиградиента для участков слева и справа от соответствующей точки могут вычисляться независимо.

Предполагаем, что такое разбиение на участки независимого построения направления спуска выполнено. Далее для участков с предельными уклонами  $b_j$  обнуляем соответствующие компоненты проекции антиградиента, а для участков с предельными значениями разностей уклонов  $b_{j+1} - b_j$ , но без предельных значений уклонов – берем среднее арифметическое соответствующих компонент антиградиента. Это простое правило следует непосредственно из определения проекции и структуры матрицы ограничений [11–13]. Если на участке с предельными значениями разностей уклонов  $b_{j+1} - b_j$  есть хотя бы один предельный уклон, то все компоненты проекции антиградиента для такого участка равны нулю. Естественно, что для свободных переменных, компоненты проекции антиградиента равны соответствующим компонентам антиградиента. При наличии высотных ограничений найденное направление спуска может оказаться недопустимым, и приходится использовать более сложные алгоритмы поиска направления спуска.

Ранее был предложен алгоритм, основанный на построении базиса в нуль-пространстве матрицы активных ограничений (матрица  $\mathbf{C}$ ) и в дополнении к нему (матрица  $\mathbf{B}$ ) без учета некоторых активных ограничений, а затем определения направления спуска при решении системы линейных уравнений, размерность которой равна числу активных ограничений, не учтенных при переходе к новому базису [14]. Каждый базисный вектор (столбец матрицы  $\mathbf{B}$ ) должен нарушать одно и только одно ограничение. Использование такого базиса позволяет не только определять направление спуска, но и легко решать вопрос об исключении ограничений из активного набора.

В нашем случае, если высотные ограничения не активны, новый базис строится

очень просто. Когда на участке активны только ограничения по разности  $b_j$  (участок сдвига в пространстве уклонов), базисная матрица  $\mathbf{C}$  состоит из единственного вектора, все компоненты которого равны единице. Число переменных на единицу больше числа активных ограничений. Вектор, меняющий первую и только первую разность  $(b_2 - b_1)$ , имеет первую компоненту нулевую, а остальные по единице. Для следующей разности имеем вектор с двумя нулевыми первыми компонентами, а остальными – равными единице и т.д. В итоге для таких участков полная базисная матрица – это нижняя треугольная матрица, ненулевые элементы которой равны 1, если разности уклонов принимают максимальные значения, и минус 1, если разности принимают минимальные значения. Легко убедиться, что тогда  $\mathbf{AC} = \mathbf{0}$  и  $\mathbf{AB} = \mathbf{E}$ , где  $\mathbf{A}$  – двухдиагональная матрица активных ограничений,  $\mathbf{E}$  – единичная матрица.

Для решения вопроса о возможности исключения ограничения по разности переменных  $b_{j+1} - b_j$  из активного набора достаточно вычислить средние арифметические компонент антиградиента на соответствующем участке от его начала до  $j$ -ой включительно ( $S_{\text{лев.}}$ ) и от  $j+1$ -ой до конца участка ( $S_{\text{прав.}}$ ). Если разность  $b_{j+1} - b_j$  принимала максимальное значение, то соответствующее ограничение можно исключить при  $S_{\text{прав.}} > S_{\text{лев.}}$ , а если она принимала минимальное значение, то это ограничение исключается при  $S_{\text{прав.}} < S_{\text{лев.}}$ .

Пусть активные ограничения на разность переменных отсутствуют, но одна или несколько смежных переменных  $b_j$  принимают предельные значения. Это участок, на котором все компоненты проекции антиградиента равны нулю (нулевой участок). Число активных ограничений равно числу переменных. Размерность нуль-пространства равна нулю, и базисная матрица  $\mathbf{C}$  отсутствует. Базисная матрица в дополнении задает полный базис. Это единичная матрица  $\mathbf{B} = \mathbf{E}$ . Если же на участке с предельной разностью смежных переменных есть одно активное ограничение на отдельную переменную, то это тоже нулевой участок. Матрица  $\mathbf{C}$  отсутствует, а в качестве матрицы  $\mathbf{B}$  в дополнении можно принять нижнюю треугольную матрицу независимо от того, какая конкретно переменная на данном участке приняла предельное значение. Если таких переменных больше, чем одна, имеем вырожденный случай, и для всех таких переменных, кроме одной, надо увеличить правую часть неравенства на малое число и тем самым исключить вырожденность. Матрицы  $\mathbf{C}$  и  $\mathbf{B}$  независимого построения проекции антиградиента строятся отдельно.

Для промежуточных свободных участков в базисную матрицу  $\mathbf{C}$  включаются векторы из канонического базиса. Построение полной базисной матрицы дает возможность построить направление спуска (приведенный антиградиент) при наличии дополнительных линейных ограничений [11]. В нашем случае речь идет о высотных ограничениях.

Обозначим матрицу дополнительных линейных ограничений через  $\mathbf{A}_1$ , тогда придется решать систему линейных уравнений с матрицей  $\mathbf{A}_1\mathbf{C}(\mathbf{A}_1\mathbf{C})^T$  ( $\mathbf{C}$  – базисная матрица нуль-пространства всех остальных ограничений). Как показано выше, построить ее очень просто. Если же число дополнительных ограничений велико, то такие вычисления на каждой итерации нежелательны.

Рассмотрим общий случай построения базиса в нуль-пространстве матрицы  $\mathbf{A}(m \times n)$ , строки которой предполагаются линейно независимыми. Алгоритм:

1. Берем канонический базис с единичной матрицей  $\mathbf{E}$ , его векторы  $\mathbf{e}_j$  ( $j = 1, 2, \dots, n$ ).

2. Берем первую строку матрицы  $\mathbf{A}$ , т.е.  $\mathbf{a}_1$ , убираем из базиса  $\mathbf{e}_1$ , а остальные векторы  $\mathbf{e}_k$  ( $k = 2, 3, \dots, n$ ) пересчитываем по формуле:

$$\mathbf{e}_k^! = \mathbf{e}_k - (\mathbf{a}_1^T, \mathbf{e}_k) / (\mathbf{a}_1^T, \mathbf{e}_1) \mathbf{e}_1$$

Все  $\mathbf{e}_k^!$  дают  $(\mathbf{e}_k^!, \mathbf{a}_1^T) = 0$ ; они линейно независимы.

3. Работаем с базисом  $\mathbf{e}_k^!$  ( $k = 2, 3, \dots, n$ ) и второй строкой матрицы  $\mathbf{A}$  по той же схеме: убираем из базиса  $\mathbf{e}_2^!$ , а остальные векторы  $\mathbf{e}_k^!$  ( $k = 3, \dots, n$ ) пересчитываем по формуле:

$$\mathbf{e}_k^{!!} = \mathbf{e}_k^! - (\mathbf{a}_2^T, \mathbf{e}_k^!) / (\mathbf{a}_2^T, \mathbf{e}_2^!) \mathbf{e}_2^!$$

Векторы с двумя штрихами дают  $(\mathbf{e}_k^{!!}, \mathbf{a}_2^T) = 0$ , но как линейная комбинация векторов с одним штрихом они дают  $(\mathbf{e}_k^{!!}, \mathbf{a}_1^T) = 0$ , то есть это базис в нуль-пространстве матрицы, составленной из двух первых строк  $\mathbf{A}$ .

4. Далее аналогично до исчерпания строк матрицы  $\mathbf{A}$  получаем  $(n - m)$  базисных векторов в нуль-пространстве матрицы  $\mathbf{A}$ .

Важно отметить, что процесс построения базисных векторов можно начать после того, как некоторые ограничения уже учтены.

В рассматриваемой задаче целесообразно построить базисную матрицу  $\mathbf{C}$ , как изложено выше, а затем поочередно подключать активные высотные ограничения, преобразуя матрицу  $\mathbf{C}$ . При этом вообще не требуется решения систем линейных уравнений на каждой итерации. Кроме того, в качестве альтернативы реализован способ учета высотных ограничений с помощью штрафных функций.

Выбирая в качестве переменных уклоны в начале каждого элемента, целевую функцию запишем в виде

$$\Phi_r(\mathbf{b}) = \lambda_r K(\mathbf{b}) + S(\mathbf{b}),$$

здесь  $K(\mathbf{b})$  – исходная целевая функция, градиент которой вычисляется пересчетом частных производных по рабочим отметкам в производные по уклонам, т.е. по новым переменным  $b_i$ ;

$S(\mathbf{b})$  – сумма квадратов невязок в высотных ограничениях, включая ограничение по конечной точке;

$\lambda_r$  – последовательность коэффициентов,  $\lim \lambda_r = 0$  ( $r \rightarrow \infty$ ).

Задачу приходится решать при нескольких последовательных значениях  $\lambda_r$ . Расчеты с заданным  $\lambda_r$  прекращаются, как только достигнута требуемая точность решения задачи. Если все высотные ограничения выполнены, счет прекращается. В противном случае  $r$  увеличивается на единицу, и счет продолжается с новым значением  $\lambda_r$ .

Исключительная простота системы ограничений и – по сравнению с проектированием «цепочки» – малая размерность задачи позволяют решать задачу оптимизации проектной линии, составленной из парабол и прямых, за малое время, если не требовать высокой точности выполнения высотных ограничений (точнее 0.01 м). В практике этого обычно достаточно.

Заметим, что при оптимизации не только уклонов, но и длин элементов усложнение алгоритмов не привело к положительным результатам. Это обусловлено тем, что длины элементов в практике принимаются кратными 10 м, а при оптимизации изменения длин

оказываются менее 10 м. Возникает задача с целочисленными переменными (длины элементов) и непрерывными переменными (уклоны), для решения которой нет математически корректных алгоритмов.

Вопрос о целесообразности оптимизации длин элементов параболического сплайна в рассматриваемой проектной задаче остается открытым, но, на наш взгляд, он не имеет первостепенной важности.

### Выводы

В системах автоматического проектирования линейных сооружений, в частности, автодорог, оптимизация продольного профиля в виде параболического сплайна с вычислительной точки зрения не представляет сложностей. Предложенные алгоритмы могут быть рекомендованы для реализации на практике.

### Литература:

1. Использование математических методов оптимизации и ЭВМ при проектировании продольного профиля железных дорог / под ред. Б.К. Маляевского. М.: Транспорт, 1977. 169 с. (Труды ВНИИ транспортного строительства. Вып. 101).
2. Скворцов А.В. Трудности перехода от автоматизированного проектирования к информационному моделированию дорог // САПР и ГИС автомобильных дорог. 2015. № 2. С. 4–12. DOI: 10.17273/CADGIS.2015.2.1.
3. ГОСТ 33100-2014 Дороги автомобильные общего пользования. Правила проектирования автомобильных дорог. Дата введения 01.02.2016.
4. Struchenkov V.I. Mathematical models and optimization in line structure routing: survey and advanced results // Int. J. Communications, Network and System Sciences. Special Issue: Models & Algorithms for Applications. 2012. № 5. P. 631–637. DOI: 10.4236/ijcns.2012.529073.
5. Кривых И.В., Мирза Н.С. Обзор зарубежных САПР автомобильных дорог // САПР и ГИС автомобильных дорог. 2015. № 2. С. 68–77. DOI: 10.17273/CADGIS.2015.2.11.
6. Bentley Rail Track [Электронный ресурс]. URL: <http://www.bentley.com/>
7. CARD/1 [Электронный ресурс]. URL: <http://www.card-1.com/en/home/>
8. Topomatic Robur [Электронный ресурс]. URL: <http://www.topomatic.ru>
9. Михалевич В.С., Шор Н.З. Математические основы решения задачи выбора оптимального очертания продольного профиля // Труды ВНИИ транспортного строительства. 1964. Вып. 51. С. 14–27.
10. Гилл Ф., Мюррей У., Райт М. Практическая оптимизация: пер. с англ. М.: Мир, 1985. 509 с.
11. Chapra S.C., Canale R.P. Numerical methods for engineers: 5th Edition. New York: McGraw-Hill, 2006. 926 p.
12. Аоки М. Введение в методы оптимизации: пер. с англ. М.: Наука, 1977. 334 с.
13. Черноруцкий И.В. Методы оптимизации. Компьютерные технологии. СПб.: БХВ-Петербург, 2011. 384 с.
14. Струченков В.И. Прикладные задачи оптимизации. М.: Солон-Пресс, 2016. 314 с.

### References:

1. The use of mathematical optimization methods and a computer in the design of the longitudinal profile of railways / Ed. by B.K. Malyavsky. Moscow: Transport Publ., 1977. 169 p. (Proceedings of the All-Union Research Institute of Transport Construction. Iss. 101). (in Russ.)
2. Skvortsov A.V. Difficulties of transition from computer-aided design to information modeling of roads // SAPR i GIS avtomobil'nykh dorog (CAD and GIS of highways). 2015. № 2. P. 4–12. (in Russ.). DOI: 10.17273/CADGIS.2015.2.1.
3. GOST 33100-2014 Road motor vehicles of general use. Rules for the design of highways. Date of introduction 01 February, 2016. (in Russ.)
4. Struchenkov V.I. Mathematical models and optimization in line structure routing: survey and advanced results // Int. J. Communications, Network and System Sciences. Special Issue: Models & Algorithms for Applications. 2012. № 5. P. 631–637. DOI: 10.4236/ijcns.2012.529073.
5. Krivykh I.V., Mirza N.S. Review of foreign CAD systems for highways // SAPR i GIS avtomobil'nykh dorog (CAD and GIS of highways). 2015. № 2. P. 68–77. (in Russ.). DOI: 10.17273/CADGIS.2015.2.11.
6. Bentley Rail Track [Electronic resource]. URL: <http://www.bentley.com/>
7. CARD/1 [Electronic resource]. URL: <http://www.card-1.com/en/home/>
8. Topomatic Robur [Electronic resource]. URL: <http://www.topomatic.ru>
9. Mikhalevich V.S., Shor N.Z. Mathematical foundations for solving the problem of choosing the optimal outline of the longitudinal profile // Proceedings of the All-Union Research Institute of Transport Construction. 1964. Iss. 51. P. 14–27. (in Russ.)
10. Gill F., Murray W., Wright M. Practical optimization. London & New York: Academic Press, 1981. 419 p.
11. Chapra S.C., Canale R.P. Numerical methods for engineers: 5th Edition. New York: McGraw-Hill, 2006. 926 p.
12. Aoki M. Introduction to optimization techniques. Los Angeles: University of California, 1971. 335 p.
13. Chernorutsky I.V. Methods of optimization. Computer technologies. Saint-Petersburg: BHV-Peterburg Publ., 2011. 384 p. (in Russ.)
14. Struchenkov V.I. Applied optimization problems. Moscow: Solon-Press Publ., 2016. 314 p. (in Russ.)

#### *Об авторе:*

*Струченков Валерий Иванович*, доктор технических наук, профессор кафедры общей информатики Института кибернетики ФГБОУ ВО «Московский технологический университет» (Россия, 119454, Москва, пр-т Вернадского, д. 78).

#### *About the author:*

*Valery I. Struchenkov*, D.Sc. (Engineering), Professor of the Chair of General Informatics, Institute of Cybernetics, Moscow Technological University (78, Vernadskogo Pr., Moscow, 119454, Russia).