

УДК 519.65

## ОБ ОДНОМ ЛОКАЛЬНО КОМОНОТОННОМ КУБИЧЕСКОМ СПЛАЙНЕ КЛАССА $C^1$

С. В. Мжачих, Ю. Н. Лапшина  
(ФГУП "РФЯЦ-ВНИИЭФ", г. Саров Нижегородской области)

В некоторых исследованиях из физических соображений требуется, чтобы кривая, интерполируя данные, была монотонной на каждом промежутке монотонности данных. Для таких задач использовать классический кубический сплайн класса  $C^2$  не всегда представляется возможным. Однако проблема решаема, причем разными путями. В работе представлен кубический сплайн класса  $C^1$ , который предназначен для решения задачи монотонной интерполяции. Сплайн будет совпадать с классическим кубическим сплайном класса  $C^2$  на участках монотонности функциональной последовательности при условии, что на этих участках классический сплайн монотонен. Отличие наблюдается лишь вблизи локальных экстремумов. Приводятся результаты численного исследования точности, при этом новый интерполянт сравнивается с другими сплайнами популярных алгоритмов.

*Ключевые слова:* кубический сплайн, монотонная интерполяция, локальная комонотонность, метод Фрича—Карлсона.

### Введение

Широкое использование сплайнов на практике ограничивается их свойством немонотонно интерполировать исходные монотонные данные. Кубический сплайн, который представлен в настоящей работе, предназначен для решения класса задач, в которых требуется монотонная интерполяция табулированных функций. Согласно терминологии современных отечественных исследователей сплайн, который будем называть трехэтапным, относится к категории локально комонотонных класса  $C^1$ . К сожалению, в разных работах можно встретить определения локальной комонотонности, которые, хоть и близки, но все же отличаются друг от друга. Будем придерживаться определения, которое сформулировано в работе [1]. В зарубежной литературе прошлого столетия такие сплайны называли локально монотонными (см., например, [2]). Об отличии между монотонными и комонотонными сплайнами можно узнать, например, из [1].

Установлено, что задача монотонной интерполяции кубическими сплайнами класса  $C^1$  всегда разрешима, т. е. при любых монотонных данных можно задать такие значения первых производных в узлах сетки, что сплайн будет монотонен. В литературе описано много подходов к решению задачи монотонной интерполяции сплайнами класса  $C^1$ . Обзор работ отечественных и зарубежных исследователей хорошо представлен в работе [3]. Достигнут прогресс и в поиске условий, которые гарантируют монотонность кубического сплайна класса  $C^2$  (см., например, [3, 4]).

В работе [2], авторами которой являются Фрич (F. Fritsch) и Карлсон (R. Carlson), впервые были сформулированы необходимые и достаточные условия локальной комонотонности сплайна и представлен метод расчета такого интерполянта. При этом сам алгоритм, который авторы назвали методом F—C, оказался прост в реализации, экономичен в расчете и достаточно точен для широкого класса задач. Благодаря этим достоинствам метод стал очень популярен. В работе [5] предложен

еще один локально комонотонный сплайн, который, в отличие от сплайна F—C, не зависит от направления обработки исходных данных. Однако полученный сплайн оказался неточным.

Многие известные специалисты по рассматриваемой проблеме согласны с тем, что при решении задачи монотонной интерполяции отталкиваться следует от классического кубического сплайна класса  $C^2$ , корректируя его в тех областях, где не выполняются условия комонотонности. При построении сплайна класса  $C^2$ , однако, приходится решать систему уравнений. Но эта система является простейшей и может быть решена очень экономичным методом прогонки. Этот подход нашел свое отражение в методе F—C, а также в работе [6], в которой представлены два алгоритма, названные авторами двухходовыми (two-sweep).

Авторы настоящей работы сочли нужным кратко представить метод F—C, так как их эксперименты с ним в итоге привели к новому работоспособному алгоритму. Метод F—C, а также второй алгоритм из работы [6], обеспечивающий максимально возможный для кубических сплайнов порядок аппроксимации, использованы при численном тестировании предлагаемого трехэтапного метода.

### 1. Классический кубический сплайн класса $C^2$

Пусть последовательность значений  $t_0, t_1, \dots, t_n$  обладает свойством  $t_{i-1} < t_i$  и каждому значению  $t_i$  соответствует единственное значение  $f_i$ , т. е.  $f_i = f(t_i)$ . Построим функцию  $p(t) \in C^2[t_0, t_n]$ , для которой  $p(t_i) = f_i$ . Для  $t \in [t_{i-1}, t_i]$ , где  $1 \leq i \leq n$ , функцию  $p(t)$  представим\* в виде полинома третьей степени

$$p_i(t) \equiv b_i(t - t_{i-1})^3 + c_i(t - t_{i-1})^2 + d_{i-1}(t - t_{i-1}) + f_{i-1}. \quad (1)$$

Из непрерывности  $p(t)$  и  $p'(t)$  ( $p_{i+1}(t_i) = p_i(t_i) = f_i$ ;  $p'_{i+1}(t_i) = p'_i(t_i) = d_i$ ) определяем коэффициенты  $b_i$  и  $c_i$ :

$$b_i = \frac{1}{\tau_i^2}(d_{i-1} + d_i - 2\delta_i); \quad c_i = \frac{1}{\tau_i}(-2d_{i-1} - d_i + 3\delta_i), \quad (2)$$

где

$$\tau_i \equiv t_i - t_{i-1}; \quad \delta_i \equiv \frac{1}{\tau_i}(f_i - f_{i-1}). \quad (3)$$

Из непрерывности  $p''(t)$  ( $p''_{i+1}(t_i) = p''_i(t_i) = 2c_{i+1}$ ) следуют уравнения

$$d_{i-1} - C_i d_i + B_i d_{i+1} = F_i, \quad i = \overline{1, n-1},$$

где

$$B_i = \frac{\tau_i}{\tau_{i+1}}; \quad C_i = -2(1 + B_i); \quad F_i = 3(\delta_i + \delta_{i+1}B_i).$$

Замыкают систему уравнений граничные соотношения (ГС)

$$d_0 = \eta_L d_1 + \varepsilon_L; \quad d_n = \eta_R d_{n-1} + \varepsilon_R. \quad (4)$$

Например, условию  $p''(t_0) = 0$ , которое принято называть естественным, отвечает задание  $c_1 = 0$ , и, следовательно,  $\eta_L = -1/2$ ;  $\varepsilon_L = 3\delta_1/2$ . Естественному условию  $p''(t_n) = 0$  отвечает задание  $3b_n \tau_n + c_n = 0$ , и, следовательно,  $\eta_R = -1/2$ ;  $\varepsilon_R = 3\delta_n/2$ .

Таким образом, для построения кубического сплайна потребуется решить систему линейных уравнений, матрица которой содержит три диагонали. Решаем систему методом прогонки. Условия  $|\eta_L| \leq 1$  и  $|\eta_R| \leq 1$  обеспечивают устойчивость вычислений.

\* Иногда будем использовать символ " $\equiv$ " для задания равенств по определению.

## 2. Необходимые и достаточные условия локальной комонотонности

Если ослабить требование к гладкости интерполирующей функции и предположить, что  $p(t) \in C^1[t_0, t_n]$ , то в выражении (2) достаточно явно задать значения  $d_i$ , например, используя трехточечную формулу, точную для параболы:

$$d_i = \gamma_i \delta_{i+1} + (1 - \gamma_i) \delta_i, \quad \gamma_i = \beta_i / (\alpha_i + \beta_i), \quad \alpha_i = \tau_{i+1}, \quad \beta_i = \tau_i.$$

При задании  $\alpha_i = |\delta_{i+2} - \delta_{i+1}|$ ,  $\beta_i = |\delta_i - \delta_{i-1}|$  получаем сплайн Акима (Н. Akima) [2].

Будем считать сплайн локально комонотонным при выполнении следующих условий на каждом отрезке  $[t_{i-1}, t_i]$  ([1]):

- если  $\delta_i = 0$ , то  $p'(t) \equiv 0$  для любой точки  $t \in [t_{i-1}, t_i]$ . Следовательно, сплайн на интервале имеет вид константы (для кубического сплайна класса  $C^1$   $d_{i-1} = d_i = 0$ );
- если  $\delta_i \neq 0$ , то для любой точки  $t \in (t_{i-1}, t_i)$  истинно неравенство  $\delta_i p'(t) \geq 0$ , т. е. знак значения  $p'(t)$  совпадает со знаком двухточечной разностной производной  $\delta_i$ . При этом для сплайна  $p(t)$  на интервале допускается точка перегиба:  $p'(t_i^*) = p''(t_i^*) = 0$ .

Заметим, что свойство неотрицательности (неположительности) комонотонных интерполянтов для неотрицательных (неположительных) данных следует автоматически, так как точки локальных экстремумов сплайна и последовательности  $f_i$  совпадают.

Сформулируем условие неотрицательности параболы

$$d(t) \equiv \tilde{b}(t - t_0)^2 + \tilde{c}(t - t_0) + d_0,$$

заданной для  $t \in [t_0, t_1]$ . Пусть  $\tau_1 \equiv t_1 - t_0$  и известны значения

$$d(t_0) = d_0 \geq 0; \quad d(t_1) = d_1 \geq 0; \quad \bar{d} \equiv \frac{1}{\tau_1} \int_{t_0}^{t_1} d(t) dt = \frac{\tilde{b}}{3} \tau_1^2 + \frac{\tilde{c}}{2} \tau_1 + d_0 > 0. \quad (5)$$

Такой параболе соответствуют коэффициенты

$$\tilde{b} = \frac{3}{\tau_1^2} (d_0 + d_1 - 2\bar{d}); \quad \tilde{c} = \frac{2}{\tau_1} (-2d_0 - d_1 + 3\bar{d}).$$

Парабола имеет глобальный минимум на интервале  $(t_0, t_1)$ , если  $d'(t_0) = \tilde{c} < 0$  и  $d'(t_1) = 2\tilde{b}\tau_1 + \tilde{c} > 0$ , т. е. при условии

$$2x + y - 3 > 0; \quad x + 2y - 3 > 0, \quad (6)$$

где  $x = d_0/\bar{d} \geq 0$  и  $y = d_1/\bar{d} \geq 0$ .

Если неравенства (6) истинны, то внутри отрезка  $[t_0, t_1]$  существует точка  $t_{\min} = -\tilde{c}/(2\tilde{b})$ , которой соответствует глобальный минимум параболы, причем

$$d(t_{\min}) = d_0 - \frac{\tilde{c}^2}{4\tilde{b}} = \frac{1}{3} \frac{M(x, y)}{2 - x - y} \bar{d}, \quad (7)$$

где

$$M(x, y) \equiv x^2 + y^2 + xy - 6x - 6y + 9 = M_+(x, y) M_-(x, y), \quad M_{\pm}(x, y) \equiv x + y - 3 \pm \sqrt{xy}. \quad (8)$$

Определим замкнутое множество

$$\mathcal{M} \equiv \{(x, y) : x \geq 0, y \geq 0, M_-(x, y) \leq 0\} \quad (9)$$

с внешней границей  $\{(x, y) : 0 \leq x \leq 3, y = 0\} \cup \{(x, y) : 0 \leq y \leq 3, x = 0\} \cup \partial\mathcal{M}$ , где

$$\partial\mathcal{M} \equiv \{(x, y) : x \geq 0, y \geq 0, M_-(x, y) = 0\}. \quad (10)$$

На рис. 1 представлены множество  $M$ , граница  $\partial M$ , ( $M_-(x, y) = 0$ ), а также графики линейных зависимостей вида  $f(x, y) = 0$  из выражений (6) и (7).

Парабола со свойствами, указанными в (5), неотрицательна на отрезке  $[t_0, t_1]$ , если хотя бы одно из неравенств (6) является ложным, так как внутри отрезка такая парабола не имеет экстремума в виде минимума. Эта область множества  $M$  закрашена на рис. 1 ярко-желтым цветом. При наличии глобального минимума неравенство  $d(t_{\min}) \geq 0$  выполняется, если точка  $(x, y)$  находится внутри эллипса  $M(x, y) = 0$ . Эта область на рис. 1 имеет светло-желтый цвет. Итак, принадлежность точки  $(x, y)$  множеству  $M$  задает условие неотрицательности параболы на отрезке  $[t_0, t_1]$ .

Теперь на отрезке  $[t_0, t_1]$  рассмотрим полином третьей степени

$$p(t) \equiv b_1(t - t_0)^3 + c_1(t - t_0)^2 + d_0(t - t_0) + f_0,$$

для которого  $p(t_0) = f_0$ ,  $p(t_1) = f_1$  и  $p'(t) \equiv d(t)$ . Нетрудно убедиться, что в этом случае

$$b_1 = \frac{\tilde{b}}{3} = \frac{d_0 + d_1 - 2\bar{d}}{\tau_1^2}; \quad c_1 = \frac{\tilde{c}}{2} = \frac{-2d_0 - d_1 + 3\bar{d}}{\tau_1}; \quad \delta_1 \equiv \frac{f_1 - f_0}{\tau_1} = b_1\tau_1^2 + c_1\tau_1 + d_0 = \bar{d}.$$

Заметим, что для расчета коэффициентов  $b_1$  и  $c_1$  используются формулы (2). Таким образом,  $p'(t) \geq 0$  на отрезке  $[t_0, t_1]$  тогда и только тогда, когда  $(d_0/\delta_1, d_1/\delta_1) \in M$ .

Предполагалось, что  $d_0 \geq 0$ ,  $d_1 \geq 0$  и  $f_1 > f_0$ . Однако, к множеству  $M$  приводит и аналогичное исследование неравенства  $p'(t) \leq 0$  для  $d_0 \leq 0$ ,  $d_1 \leq 0$  и  $f_1 < f_0$ .

Впервые множество  $M$  было представлено в работе [2], согласно которой кубический сплайн  $p(t) \in C^1[t_0, t_n]$ , заданный выражениями (1)–(3), будет локально комонотонным, если для всех отрезков  $[t_{i-1}, t_i]$   $\delta_i \neq 0$  и выполнено условие  $(x_i, y_i) \in M$ . Здесь  $x_i = d_{i-1}/\delta_i$ ,  $y_i = d_i/\delta_i$ ,  $1 \leq i \leq n$ .

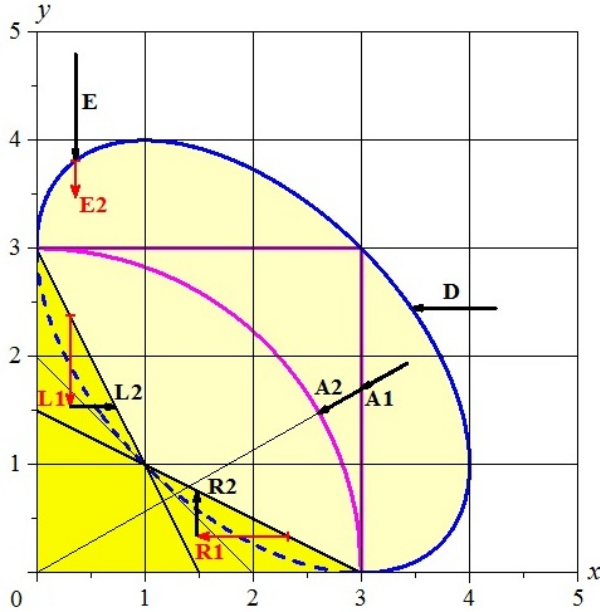


Рис. 1. Множество локальной комонотонности кубического сплайна класса  $C^1$  и подмножества Фрича–Карлсона:  $\square$  — множество, на котором сплайн не является комонотонным;  $\blacksquare$ ,  $\text{light yellow}$  — множество  $M$  локальной комонотонности сплайна;  $\text{blue}$  — график  $M_-(x, y) = 0$  (граница  $\partial M$ ),  $\text{dashed blue}$  — график  $M_+(x, y) = 0$

### 3. Метод F–C

**3.1. Расчет начальных данных.** Прежде всего, для внутренних точек  $t_i$  ( $1 \leq i < n$ ) проверяем неравенство  $\delta_i \delta_{i+1} \leq 0$  и в случае его истинности задаем  $d_i = 0$ .

Далее будем рассматривать подмножества из граничащих между собой отрезков  $[t_{i-1}, t_i]$ , для которых подпоследовательность  $f_i$  ведет себя строго монотонно, т. е. для всех отрезков подмножества либо  $\delta_i < 0$ , либо  $\delta_i > 0$ .

Для внутренних точек  $t_i$  строго возрастающей подпоследовательности  $f_i$  проводим коррекцию данных сплайна класса  $C^2$  с помощью оператора присваивания  $d_i \leftarrow \max(0, d_i)$ . Для строго убывающей подпоследовательности используем оператор  $d_i \leftarrow \min(0, d_i)$ . Итак, для всех внутренних точек подпоследовательности истинны неравенства  $d_i \delta_i \geq 0$ .

При задании ГС вида  $p''(t_0) = 0$  пересчитаем значение  $d_0$  по формулам (4). В случае  $d_0 \delta_1 < 0$  положим  $d_0 = 0$  и определим новое значение  $d_1$  из первого соотношения (4): получим  $d_1 = 3\delta_1$ . Заметим, что  $d_1 \delta_1 > 0$ , а отрезку  $[t_0, t_1]$  соответствует точка  $(0, 3)$  на рис. 1.

Аналогично обрабатываются переменные  $d_{n-1}$  и  $d_n$  при задании ГС вида  $p''(t_n) = 0$ .

**3.2. Алгоритм.** Для построения локально комонотонного сплайна авторами статьи [2] на множестве  $\mathcal{M}$  определяется его замкнутое подмножество  $\mathcal{J}$  со свойством

$$(\tilde{x}, \tilde{y}) \in \mathcal{J}, \text{ если } \tilde{x} \in [0, x], \tilde{y} \in [0, y] \text{ и } (x, y) \in \mathcal{J}. \quad (11)$$

То есть любая точка такого подмножества при уменьшении значений  $x$  или  $y$  никогда не выйдет за его пределы. Как видно из рис. 1, условие (11) выполнимо не для всех точек множества  $\mathcal{M}$ . В работе [2] рекомендовано использовать в качестве множества  $\mathcal{J}$  либо множество квадрата, либо множество квадранта круга (см. рис. 1):

$$\mathcal{J}_1 \equiv \{(x, y) : 0 \leq x \leq 3; 0 \leq y \leq 3\}; \quad \mathcal{J}_2 \equiv \{(x, y) : 0 \leq x \leq 3; 0 \leq y \leq \sqrt{9 - x^2}\}.$$

Используем процедуру прямого перебора отрезков  $[t_{i-1}, t_i]$ . Вообще говоря, алгоритм можно адаптировать и к обратному ходу, т. е. меняя индекс  $i$  от  $n$  до 1, но тогда получим другой сплайн.

Если  $(x_i, y_i) \notin \mathcal{J}$ , то точка проецируется на границу  $\partial\mathcal{J}$  в направлении начала координат (см. стрелки A1 и A2 на рис. 1). Алгоритм может уменьшить как значение  $y_i$ , так и значение  $x_i$ . Уменьшение  $x_i$  автоматически приведет к уменьшению ранее определенного значения  $y_{i-1}$ . Однако предыдущая точка после пересчета не покинет множество  $\mathcal{J}$  в силу его свойства. Результатом выполнения алгоритма будут значения коэффициентов  $d_i$ , при этом для всех точек справедливо условие  $(x_i, y_i) \in \mathcal{J}$ . Затем по формулам (2) определяются другие коэффициенты сплайна.

Авторы метода не осветили вопрос применения алгоритма к крайним отрезкам. Сделаем это сами. Для этого потребуются функции

$$R(x) \equiv -\frac{1}{2}x + \frac{3}{2}; \quad L(y) \equiv -\frac{1}{2}y + \frac{3}{2}; \quad G(s) \equiv \frac{1}{2} \left[ 6 - s + \sqrt{3s(4-s)} \right] \quad (12)$$

и рис. 1.

Двухточечное ГС вида (4) задает зависимость  $y_n = \eta_R x_n + \varepsilon_R / \delta_n$ . Ограничимся рассмотрением ГС вида  $p''(t_n) = 0$ , которому отвечает линейная зависимость  $y = R(x)$ . Исходим из того, что точка  $(x_n, y_n)$ , построенная по данным сплайна класса  $C^2$  (даже с учетом его коррекции), находится на прямой линии  $y = R(x)$  и принадлежит множеству  $\mathcal{J}$ . После обработки точки  $(x_{n-1}, y_{n-1})$  значение  $x_n$  может только уменьшиться (см. стрелку R1). Алгоритм проецирует точку  $(x_n, y_n)$  на линию  $y = R(x)$  с сохранением значения  $x_n$  (см. стрелку R2), т. е. не затрагивая точки слева. Значение  $y_n$  при этом увеличивается, но точка  $(x_n, y_n)$  остается внутри множества  $\mathcal{J}$ .

Аналогично рассматривается ГС вида  $p''(t_0) = 0$ . Изначально точка  $(x_1, y_1)$  находится на линии  $x = L(y)$  и  $(x_1, y_1) \in \mathcal{J}$ . После обработки отрезка  $[t_1, t_2]$  значение  $y_1$  может уменьшиться (см. стрелку L1). Полученную в результате точку  $(x_1, y_1)$  всегда можно спроецировать на линию  $x = L(y)$  с сохранением значения  $y_1$  (см. стрелку L2), т. е. не затрагивая точки справа.

Одноточечное ГС вида  $p'(t_n) = d_n = \varepsilon_R$  фиксирует значение  $y_n = \varepsilon_R / \delta_n$ . В частном случае  $\varepsilon_R = 0$ . Это значение соответствует сплайну класса  $C^2$ , который служит базисом для конструирования. Пусть  $(x_n, y_n) \notin \mathcal{M}$ , т. е.  $M_-(x_n, y_n) > 0$ . Тогда проецируем точку  $(x_n, y_n)$  на границу  $\mathcal{M}$ , смещаясь влево по линии  $y = y_n$  (см. стрелку D). В случае  $y_n \leq 3$  задача всегда имеет решение  $x_n = G(y_n)$ . Уменьшение значения  $x_n$  автоматически уменьшает значение  $y_{n-1}$ , что оставляет точку  $(x_{n-1}, y_{n-1})$  внутри множества  $\mathcal{J}$ .

Рассматривая ГС вида  $p'(t_0) = d_0 = \varepsilon_L$ , имеем  $x_1 = \varepsilon_L / \delta_1$ . В частном случае  $\varepsilon_L = 0$ . Пусть  $(x_1, y_1) \notin \mathcal{M}$ , т. е.  $M_-(x_1, y_1) > 0$ . Тогда проецируем точку  $(x_1, y_1)$  на границу  $\mathcal{M}$ , смещаясь вниз

по линии  $x = x_1$  (см. стрелку E). В случае  $x_1 \leq 3$  задача всегда имеет решение  $y_1 = G(x_1)$ . После обработки соседней точки  $(x_2, y_2)$  значение  $y_1$  может только уменьшиться (см. стрелку E2), но это не приведет к выходу точки  $(x_1, y_1)$  из множества  $\mathcal{J}$ .

#### 4. Трехэтапный алгоритм

**4.1. Множества.** Множества  $\mathcal{M}$  и  $\partial\mathcal{M}$  были описаны в разд. 2 (см. (8)–(10)). Для представления трехэтапного алгоритма понадобятся новые множества:

$$\begin{aligned} \mathcal{G} &\equiv \{(x, y) : 1 < x < 4, G(x) < y\} \cup \{(x, y) : 4 \leq x, 1 < y\}; \\ \mathcal{M}_x &\equiv \{(x, y) : 0 \leq y < 1, G(y) < x \leq 4\}; \quad \mathcal{M}_y \equiv \{(x, y) : 0 \leq x < 1, G(x) < y \leq 4\}; \\ \mathcal{F}_x &\equiv \{(x, y) : 4 < x, 0 \leq y \leq 1\}; \quad \mathcal{F}_y \equiv \{(x, y) : 0 \leq x \leq 1, 4 < y\} \end{aligned}$$

и рис. 2.

Определим граничные множества:

– для множеств  $\mathcal{M}_y$  и  $\mathcal{M}$ :  $\partial\mathcal{M}_y \equiv \{(x, y) : 0 \leq x \leq 1, y = G(x)\}, \quad \partial\mathcal{M}_y \subset \mathcal{M}, \quad \partial\mathcal{M}_y \not\subset \mathcal{M}_y; \quad (13)$

– для множеств  $\mathcal{M}_x$  и  $\mathcal{M}$ :  $\partial\mathcal{M}_x \equiv \{(x, y) : 0 \leq y \leq 1, x = G(y)\}, \quad \partial\mathcal{M}_x \subset \mathcal{M}, \quad \partial\mathcal{M}_x \not\subset \mathcal{M}_x; \quad (14)$

– для множеств  $\mathcal{G}$  и  $\mathcal{M}$ :  $\partial\mathcal{G} \equiv \{(x, y) : 1 \leq x \leq 4, y = G(x)\}, \quad \partial\mathcal{G} \subset \mathcal{M}, \quad \partial\mathcal{G} \not\subset \mathcal{G}; \quad (15)$

– для множеств  $\mathcal{F}_y$  и  $\mathcal{M}_y$ :  $\partial\mathcal{F}_y \equiv \{(x, y) : 0 \leq x \leq 1, y = 4\}, \quad \partial\mathcal{F}_y \subset \mathcal{M}_y, \quad \partial\mathcal{F}_y \not\subset \mathcal{F}_y;$

– для множеств  $\mathcal{F}_x$  и  $\mathcal{M}_x$ :  $\partial\mathcal{F}_x \equiv \{(x, y) : x = 4, 0 \leq y \leq 1\}, \quad \partial\mathcal{F}_x \subset \mathcal{M}_x, \quad \partial\mathcal{F}_x \not\subset \mathcal{F}_x;$

$$\partial\mathcal{M} = \partial\mathcal{M}_x \cup \partial\mathcal{G} \cup \partial\mathcal{M}_y, \quad \partial\mathcal{M} \subset \mathcal{M}; \quad \partial\mathcal{J} \equiv \partial\mathcal{F}_x \cup \partial\mathcal{G} \cup \partial\mathcal{F}_y, \quad \partial\mathcal{J} \subset \mathcal{J}.$$

Данные для предлагаемого алгоритма готовятся так же, как для метода F–C (см. разд. 3.1). Далее полагаем, что на всем отрезке  $[t_0, t_n]$  последовательность  $f_i$  строго монотонна.

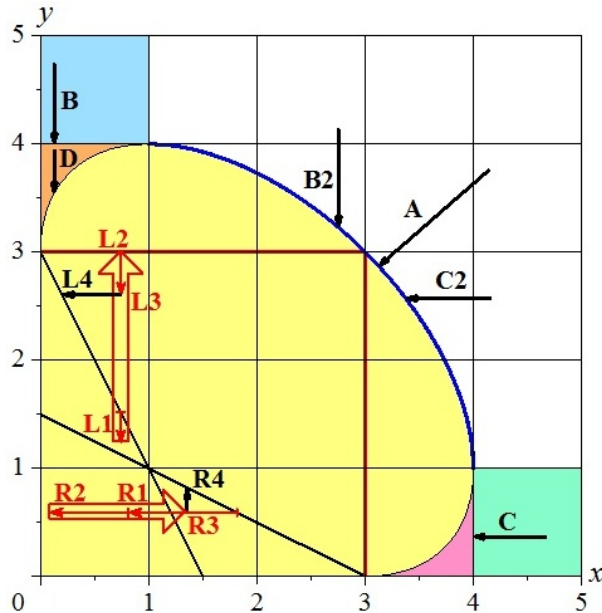


Рис. 2. Множества для трехэтапного алгоритма расчета локально комонотонного сплайна:  $\square$  – открытое множество  $\mathcal{G}$ ;  $\blacksquare$  – замкнутое множество  $\mathcal{M}$ ;  $\blacksquare$  – множество  $\mathcal{M}_x$ ;  $\blacksquare$  – множество  $\mathcal{M}_y$ ;  $\blacksquare$  – множество  $\mathcal{F}_x$ ;  $\blacksquare$  – множество  $\mathcal{F}_y$ ;  $\text{—}$  – граница  $\partial\mathcal{G}$

**4.2. Этап 1.** На этом этапе точки  $(x_i, y_i)$ , не принадлежащие множеству  $\mathcal{J}$ , проецируются на него. Под проецированием на множество подразумеваем построение нормали к его границе. Порядок действий на начальном этапе следующий:

1. Последовательно рассматриваются все отрезки  $[t_{i-1}, t_i]$ . Направление обработки значения не имеет. Рассчитаем расстояние  $l_i$  от точки  $(x_i, y_i)$  до границы  $\partial\mathcal{J}$ . Полагаем  $l_i = 0$ , если

$(x_i, y_i) \in \mathcal{J}$ . В случае  $(x_i, y_i) \notin \mathcal{J}$  используем формулу евклидовой метрики

$$l_i = \sqrt{(x_i^* - x_i)^2 + (y_i^* - y_i)^2}, \quad (16)$$

где  $(x_i^*, y_i^*)$  — точка проекции. Если  $(x_i, y_i) \in \mathcal{F}_x$ , то  $x_i^* = 4$ ,  $y_i^* = y_i$ . Если  $(x_i, y_i) \in \mathcal{F}_y$ , то  $x_i^* = x_i$ ,  $y_i^* = 4$ . Расчет точки проекции на границе  $\partial\mathcal{G}$  будет рассмотрен отдельно. На рис. 2 показаны примеры построения нормали к границе  $\partial\mathcal{J}$  в виде стрелок А, В, С.

Отметим особенность расчета точек проекции для крайних интервалов, в которых заданы одноточечные ГС.

Исследуем ГС вида  $p'(t_n) = \varepsilon_R$ . Тогда имеется фиксированное значение  $y_n = \varepsilon_R/\delta_n$ . Будем считать, что задача построения локально комонотонного сплайна при данном условии имеет решение только в случае  $y_n \leq 3$ . Если  $y_n \in (1, 3]$  и  $M_-(x_n, y_n) > 0$ , то проецирование на границу  $\partial\mathcal{G}$  проводится не по нормали, а с сохранением значения  $y_n$ , т. е.  $y_n^* = y_n$  и  $x_n^* = G(y_n^*)$  (см. на рис. 2 стрелку С2). Если  $(x_n, y_n) \in \mathcal{F}_x$ , то для точки проекции на границе  $\partial\mathcal{F}_x$  имеем  $y_n^* = y_n$ ,  $x_n^* = 4$  (см. стрелку С).

ГС вида  $p'(t_0) = \varepsilon_L$  задает значение  $x_1 = \varepsilon_L/\delta_1$ , которое нужно сохранять при проецировании. Задача построения сплайна решается только в случае  $x_1 \leq 3$ . При этом используется алгоритм, который по сути не отличается от того, что был рассмотрен в предыдущем абзаце. Если  $x_1 \in (1, 3]$  и  $M_-(x_1, y_1) > 0$ , то получаем  $x_1^* = x_1$  и  $y_1^* = G(x_1^*)$ . Действие алгоритма иллюстрируется на рис. 2 вертикальными стрелками В и В2.

2. Определяем индекс  $m$ , которому соответствует значение  $l_m^2 = \max_i l_i^2$ .
3. Если  $(x_m, y_m) \notin \mathcal{J}$ , т. е.  $l_m^2 > 0$ , то проецируем точку  $(x_m, y_m)$  на границу  $\partial\mathcal{J}$ . Получаем  $x_m = x_m^*$  и  $y_m = y_m^*$ . Далее полагаем  $l_m^2 = 0$ .
4. Пересчитаем левую точку  $(x_{m-1}, y_{m-1})$  при условии  $m > 1$  и правую точку  $(x_{m+1}, y_{m+1})$  при условии  $m < n$ . Определим новые значения  $l_k^2$  для  $k = m \pm 1$ , если  $1 \leq k \leq n$  и  $(x_k, y_k) \notin \mathcal{J}$ . При условии  $(x_k, y_k) \in \mathcal{J}$  полагаем  $l_k^2 = 0$ .
5. Повторяем алгоритм от п. 2 до выполнения условия  $l_m^2 = 0$ . Теперь все точки принадлежат множеству  $\mathcal{J}$ . Такое событие обязательно наступит, так как множество  $\mathcal{J}$  обладает свойством (11). Заметим, что каждая точка обрабатывается только один раз.

Для понимания работы алгоритма на следующих этапах потребуется знание положения крайних точек  $(x_i, y_i)$ , для которых заданы двухточечные ГС.

Рассмотрим случай задания ГС вида  $p''(t_0) = 0$ . Будем исходить из того, что до старта данного этапа  $x_1 = L(y_1)$  и  $(x_1, y_1) \in \mathcal{M}$ , т. е. изначально  $l_1 = 0$ . При обработке точки  $(x_2, y_2)$  значения  $x_2$  и, следовательно,  $y_1$  могут только уменьшаться (см. стрелку L1). Но новая точка  $(x_1, y_1)$  не выйдет из множества  $\mathcal{J}$ , и следовательно, для нее  $l_1 = 0$ .

Аналогично исследуется поведение точки  $(x_n, y_n)$  в случае задания ГС вида  $p''(t_n) = 0$ . До старта данного этапа  $y_n = R(x_n)$  и  $(x_n, y_n) \in \mathcal{M}$ , т. е. изначально  $l_n = 0$ . При обработке точки  $(x_{n-1}, y_{n-1})$  значения  $y_{n-1}$  и, следовательно,  $x_n$  могут только уменьшаться (см. стрелку R1). Замечаем, что новая точка  $(x_n, y_n)$  не выходит из множества  $\mathcal{J}$ , и следовательно, для нее  $l_n = 0$ .

**4.3. Этап 2.** На этапе 2 точки  $(x_i, y_i)$  из множества  $\mathcal{M}_y$  переносятся в множество  $\mathcal{M} \cup \mathcal{M}_x$ . При этом индекс  $i$  меняется от 1 до  $n$ , т. е. используется прямой ход. Если  $(x_i, y_i) \in \mathcal{M}_y$ , то проецирование точки по нормали к границе  $\partial\mathcal{M}_y$  приводит к возрастанию значения  $x_i$  и убыванию значения  $y_i$ . Проводим операцию смещения точки  $(x_i, y_i)$  в направлении нормали к границе  $\partial\mathcal{M}_y$  до наступления одного из событий:

- а) текущая точка  $(x_i, y_i)$  вышла на границу  $\partial\mathcal{M}_y$ , принадлежащую области  $\mathcal{M}$ ;
- б)  $i > 1$  и точка слева  $(x_{i-1}, y_{i-1})$  при автоматическом возрастании значения  $y_{i-1}$  вышла на границу  $\partial\mathcal{M}_y \cup \partial\mathcal{G}$ , принадлежащую области  $\mathcal{M}$ .

Если событие  $b$  предшествует событию  $a$ , т. е. для новой точки  $M_-(x_i, y_i) > 0$ , то уменьшаем новое значение  $y_i$  вертикальным сдвигом вниз (вдоль оси  $Oy$ ) до выхода точки  $(x_i, y_i)$  на границу  $\partial M_y$ . Последняя операция не меняет положения точки слева.

Отметим две особенности выполнения алгоритма на этапе 2.

Первая особенность связана с заданием ГС вида  $p''(t_0) = 0$ . Непосредственная обработка точки  $(x_1, y_1)$  оставляет ее на месте, так как  $(x_1, y_1) \notin M_y$ . Но при обработке следующей точки  $(x_2, y_2)$  в случае ее принадлежности множеству  $M_y$  с ростом значения  $x_2$  (при автоматическом возрастании значения  $y_1$ ) будет отслеживаться событие  $\bar{b}$ , которое отличается от нормы. Нерегулярное событие  $\bar{b}$  определяется выходом точки  $(x_1, y_1)$  на границу по линии  $y = 3$  при ее вертикальном сдвиге вверх (см. на рис. 2 стрелку L2).

Вторая особенность связана с заданием ГС вида  $p'(t_0) = \varepsilon_L$ . В этом случае имеется не подлежащее коррекции значение  $x_1 = \varepsilon_L / \delta_1$ . Следовательно, если  $(x_1, y_1) \in M_y$ , то для проецирования точки на границу  $\partial M_y$  используем вертикальное смещение вниз (см. стрелку D). Обработка точки  $(x_2, y_2)$  проводится обычным образом, при этом переменная  $x_1$  не затрагивается.

После обработки точки  $(x_i, y_i)$  значение  $y_i$  может уменьшиться, при этом точка  $(x_{i+1}, y_{i+1})$  сдвинется горизонтально влево и может, вообще говоря, переместиться из множества  $M$  в множество  $M_y$ , которое теперь содержит только точки  $(x_k, y_k)$  для  $k > i$ .

Исследуем поведение точки  $(x_n, y_n)$  на этапе 2 алгоритма при задании ГС вида  $p''(t_n) = 0$ . Как уже было сказано, обработка точки  $(x_{n-1}, y_{n-1})$ , которая является предпоследней в цикле, может привести к горизонтальному сдвигу влево точки  $(x_n, y_n)$  (см. стрелку R2, которая является продолжением стрелки R1 — вектора этапа 1). При непосредственной обработке точки  $(x_n, y_n)$  на этапе 2 ее положение не изменится, так как  $(x_n, y_n) \notin M_y$ .

Пусть для точки  $(x_n, y_n)$  задано ГС вида  $p'(t_n) = \varepsilon_R$ . После обработки точки  $(x_{n-1}, y_{n-1})$  точка  $(x_n, y_n)$ , сместившись влево, не сможет попасть в множество  $M_y$ , так как после этапа 1  $y_n \in [0, 3]$ . Следовательно, алгоритм оставит точку  $(x_n, y_n)$  на месте.

Итак, после этапа 2 одноточечные ГС всегда выполняются, тогда как двухточечные ГС могут не выполняться, но при этом  $y_1 \in [0, 3]$  и  $x_n \in [0, 3]$ . Далее будет показано, что это заключение справедливо и для этапа 3.

**4.4. Этап 3.** Данный этап аналогичен этапу 2, но здесь переносятся точки  $(x_i, y_i)$  из множества  $M_x$  в множество  $M$ . Используем обратный ход, т. е. индекс  $i$  меняется от  $n$  до 1. Если  $(x_i, y_i) \in M_x$ , то проецирование точки по нормали к границе  $\partial M_x$  приводит к увеличению значения  $y_i$  и уменьшению значения  $x_i$ . Проводим операцию смещения точки  $(x_i, y_i)$  в направлении нормали к границе  $\partial M_x$  до наступления одного из событий:

- а) текущая точка  $(x_i, y_i)$  вышла на границу  $\partial M_x$ , принадлежащую области  $M$ ;
- б)  $i < n$  и точка справа  $(x_{i+1}, y_{i+1})$  при автоматическом увеличении значения  $x_{i+1}$  вышла на границу  $\partial M_x \cup \partial \mathcal{G}$ , принадлежащую области  $M$ .

Если событие  $b$  предшествует событию  $a$ , т. е. для новой точки  $M_-(x_i, y_i) > 0$ , то уменьшаем новое значение  $x_i$  горизонтальным сдвигом влево (вдоль оси  $Ox$ ) до выхода точки на границу  $\partial M_x$ . Последняя операция не меняет положения точки справа.

Отметим две особенности выполнения алгоритма на этапе 3.

Первая особенность связана с заданием ГС вида  $p''(t_n) = 0$ . Алгоритм оставляет точку  $(x_n, y_n)$  на месте, так как  $(x_n, y_n) \notin M_x$ . Но при обработке следующей точки  $(x_{n-1}, y_{n-1})$  в случае ее принадлежности множеству  $M_x$  с ростом значения  $y_{n-1}$  (при автоматическом возрастании значения  $x_n$ ) будет отслеживаться событие  $\bar{b}$ , которое отличается от нормы. Нерегулярное событие  $\bar{b}$  определяется выходом точки  $(x_n, y_n)$  на границу по линии  $x = 3$  при ее горизонтальном сдвиге вправо (см. на рис. 2 стрелку R3).

Вторая особенность связана с заданием ГС вида  $p'(t_n) = \varepsilon_R$ . В этом случае имеем не подлежащее коррекции значение  $y_n = \varepsilon_R / \delta_n$ . Следовательно, если  $(x_n, y_n) \in M_x$ , то для проецирования точки на границу  $\partial M_x$  используем горизонтальное смещение влево. Обработка следующей точки  $(x_{n-1}, y_{n-1})$  проводится обычным образом, при этом не затрагивается переменная  $y_n$ .



После обработки точки  $(x_i, y_i)$  значение  $x_i$  уменьшится, следовательно, точка  $(x_{i-1}, y_{i-1})$  сдвинется вертикально вниз и может, вообще говоря, переместиться из множества  $\mathcal{M}$  в множество  $\mathcal{M}_x$ , которое теперь содержит только точки  $(x_k, y_k)$  для  $k < i$ . Очевидно, множество  $\mathcal{M}_y$  остается свободным от точек.

Рассмотрим поведение точки  $(x_1, y_1)$  на этапе 3 алгоритма в случае задания ГС вида  $p''(t_0) = 0$ . Как уже было сказано, обработка точки  $(x_2, y_2)$ , которая рассматривается предпоследней в цикле, приведет к вертикальному сдвигу вниз точки  $(x_1, y_1)$  (см. стрелку L3). При дальнейшем рассмотрении точки  $(x_1, y_1)$  на этапе 3 ее положение не изменится, так как  $(x_1, y_1) \notin \mathcal{M}_x$ .

Пусть для точки  $(x_1, y_1)$  задано ГС вида  $p'(t_0) = \varepsilon_L$ . После этапа 2 имеем  $x_1 \in [0, 3]$ . Следовательно, после обработки точки  $(x_2, y_2)$  новая точка  $(x_1, y_1)$  не может принадлежать множеству  $\mathcal{M}_x$  и алгоритм на этапе 3 оставляет ее без изменения.

**4.5. Восстановление двухточечных ГС.** Итак, после этапа 3 все точки будут принадлежать области  $\mathcal{M}$  и для решения задачи построения локально комонотонного сплайна класса  $C^1$  следует при необходимости восстановить двухточечные ГС, выполнив следующий алгоритм:

1. Если  $\delta_1 \neq 0$ , то при задании ГС вида  $p''(t_0) = 0$  проецируем точку  $(x_1, y_1)$  на линию  $x = L(y)$  горизонтальным смещением. В зависимости от начального положения точки  $(x_1, y_1)$  проецирование может проходить либо с увеличением, либо с уменьшением значения  $x_1$  (см. на рис. 2 стрелку L4). Заметим, что проецирование всегда возможно, так как  $y_1 \in [0, 3]$ , и оно не затрагивает соседнюю точку  $(x_2, y_2)$ . Если  $\delta_1 = 0$ , то имеем  $p'(t_0) = p''(t_0) = 0$ .
2. Если  $\delta_n \neq 0$ , то при задании ГС вида  $p''(t_n) = 0$  проецируем точку  $(x_n, y_n)$  на линию  $y = R(x)$  вертикальным смещением. В зависимости от начального положения точки  $(x_n, y_n)$  проецирование может проходить либо с увеличением (см. стрелку R4), либо с уменьшением значения  $y_n$ . Заметим, что проецирование всегда возможно, так как  $x_n \in [0, 3]$ , и оно не затрагивает соседнюю точку  $(x_{n-1}, y_{n-1})$ . Если  $\delta_n = 0$ , то имеем  $p'(t_n) = p''(t_n) = 0$ .

**4.6. Расчет точки проекции на границе эллипса.** Пусть  $(x_i^*, y_i^*) \in \partial\mathcal{G}$ . В формуле (16) вместо переменной  $y_i^*$  подставим зависимость  $y(x_i^*)$  из выражения (15) и найдем точку минимума функции  $l_i(s)$  из условия  $l_i'(s) = H(s) = 0$ , где

$$H(s) \equiv 4(s - x_i) \sqrt{3s(4 - s)} + \left[6 - s - 2y_i + \sqrt{3s(4 - s)}\right] \left[6 - 3s - \sqrt{3s(4 - s)}\right]. \quad (17)$$

Решение уравнения  $H(s) = 0$  сводится к поиску корней уравнения 4-й степени

$$s^4 + u_3 s^3 + u_2 s^2 + u_1 s + u_0 = 0, \quad (18)$$

где

$$u_3 = 2\beta - \alpha - 10; \quad u_2 = \beta^2 - 14\beta - \alpha\beta + 4\alpha + \alpha^2 + 36; \quad (19)$$

$$u_1 = -4\beta^2 + 27\beta + 4\alpha\beta - 4\alpha^2 - 54; \quad u_0 = 3(\beta - 3)^2; \quad (20)$$

$$\alpha = x_i; \quad \beta = y_i; \quad x_i^* = s; \quad y_i^* = G(s). \quad (21)$$

К задаче (18)–(21) приходим и в случае построения нормали к границе  $\partial\mathcal{M}_y$  (см. (13)). В задаче построения нормали к границе  $\partial\mathcal{M}_x$  (см. (14)) меняем значения  $x$  и  $y$  местами, т. е. полагаем

$$\alpha = y_i; \quad \beta = x_i; \quad y_i^* = s; \quad x_i^* = G(s). \quad (22)$$

Для решения уравнения 4-й степени воспользуемся методом Феррари [7], согласно которому решение уравнения (18) зависит от действительного корня его кубической резольвенты

$$z^3 + a_2 z^2 + a_1 z + a_0 = 0, \quad (23)$$

где

$$a_2 = -u_2; \quad a_1 = u_3 u_1 - 4u_0; \quad a_0 = (4u_2 - u_3^2) u_0 - u_1^2.$$

Известно, что уравнение с действительными коэффициентами нечетной степени всегда имеет хотя бы один действительный корень. Кубическое уравнение, записанное в каноническом виде (23), будем решать методом Кардано [7]. Сделаем замену  $z = t - \nu$ , где

$$\nu = a_2/3,$$

и получим приведенное уравнение

$$t^3 + pt + q = 0 \quad (24)$$

с коэффициентами

$$p = a_1 - 3\nu^2, \quad q = a_0 - \nu(p + \nu^2).$$

Уравнение 3-й степени (24) имеет три корня  $(t_1, t_2, t_3)$ , два из которых могут быть сопряженными комплексными. Число действительных корней зависит от значения дискриминанта  $D$ , где

$$D = g^3 + h^3, \quad g = p/3, \quad h = q/2.$$

В случае  $D > 0$  существует только один действительный корень. Если  $D \leq 0$ , то все три корня действительные. В случае  $D = 0$  имеем дело с кратным корнем, причем если  $p = q = 0$ , то  $t_1 = t_2 = t_3 = 0$ .

Для метода Феррари достаточно знать один действительный корень  $z = z_1$  уравнения (23). Определяем

$$z_1 = t_1 - \nu.$$

Если  $q = 0$  ( $|q| \leq \sigma$ , где  $\sigma$  — малый параметр), имеем  $t_1 = 0$ .

Если  $p = 0$  ( $|p| \leq \sigma, |q| > \sigma$ ), интересен только корень  $t_1 = -|q|^{1/3} \text{sign } q$ .

В других случаях ( $|p| > \sigma, |q| > \sigma$ ) используем

$$t_1 = \begin{cases} -2R \operatorname{ch} \left( \frac{1}{3} \operatorname{arch} \frac{h}{R^3} \right), & \text{если } D > 0 \text{ и } p < 0; \\ -2R \operatorname{sh} \left( \frac{1}{3} \operatorname{arsh} \frac{h}{R^3} \right), & \text{если } D > 0 \text{ и } p > 0; \\ -2R \cos \left( \frac{1}{3} \operatorname{arccos} \frac{h}{R^3} \right), & \text{если } D \leq 0, \end{cases} \quad R = \sqrt{|g|} \operatorname{sign } q.$$

Для уравнения 4-й степени число действительных корней кратно двум либо таких корней нет, но это не актуально для рассматриваемой задачи. Согласно методу Феррари действительные корни уравнения (18) находятся из решения двух квадратных уравнений

$$s^2 - \xi_1 s + \frac{1}{2}(z_1 - \eta) = 0; \quad s^2 - \xi_2 s + \frac{1}{2}(z_1 + \eta) = 0, \quad (25)$$

где

$$\xi_1 = \lambda - w; \quad \xi_2 = -\lambda - w; \quad w = \frac{u_3}{2}; \quad \lambda = \sqrt{w^2 - u_2 + z_1}; \quad \eta = \frac{w z_1 - u_1}{\lambda}.$$

Рассчитаем дискриминант первого уравнения из (25) по формуле

$$D_1 = \xi_1^2 - 2(z_1 - \eta).$$

Если  $D_1 \geq 0$ , то, задав  $k = 1$ , получаем действительные корни

$$s_k^\pm = \frac{1}{2} \left( \xi_k \pm \sqrt{D_k} \right). \quad (26)$$

В случае  $D_k = 0$  имеем дело с кратным корнем.

Единственный корень, который вызывает интерес, является решением уравнения  $H(s) = 0$ . Прежде всего среди найденных корней оставим только те, которые принадлежат отрезку  $[0, 4]$ . С

помощью выражений (12) и (21) преобразуем формулу (17) к виду, из которого следует, что искомый корень удовлетворяет неравенству

$$\left| (x_i^* - x_i) (2y_i^* + x_i^* - 6) - (y_i^* - y_i) (2x_i^* + y_i^* - 6) \right| \leq \sigma.$$

Неравенство справедливо и для точки проекции на границе  $\partial\mathcal{M}_x$  (см. (22)).

Если среди корней первого уравнения из (25) нет искомого, то он должен быть среди корней второго уравнения из (25), которые задаются выражением (26), где  $k = 2$  и

$$D_2 = \xi_2^2 - 2(z_1 + \eta).$$

Заметим, что  $D_2 \geq 0$ , так как действительный корень уравнения  $H(s) = 0$  существует.

## 5. Численное тестирование

Для иллюстрации применения нового метода построения локально комонотонного сплайна класса  $C^1$  были подобраны несколько тестов. Расчеты проводились с естественными граничными условиями.

В представленных ниже рисунках и таблицах используются следующие обозначения:  $C^2$  — классический кубический сплайн класса  $C^2$ ;  $F-C(\mathcal{J}_1)$ ,  $F-C(\mathcal{J}_2)$  — сплайны класса  $C^1$ , построенные с помощью метода  $F-C$  [2], где в качестве множества  $\mathcal{J}$  используется либо квадрат ( $\mathcal{J}_1$ ), либо квадрант круга ( $\mathcal{J}_2$ );  $E-J-L$  — сплайн класса  $C^1$ , построенный с помощью расширенного двухходового метода из работы [6];  $M-L$  — новый сплайн класса  $C^1$ , построенный трехэтапным методом.

Сплайны  $F-C(\mathcal{J}_1)$  и  $F-C(\mathcal{J}_2)$  в рассмотренных тестах имеют минимальные отличия и визуально совпадают, поэтому на рисунках демонстрируется только сплайн  $F-C(\mathcal{J}_1)$ . Кроме графических данных, приводятся численные значения отклонений сплайнов класса  $C^1$  от сплайна класса  $C^2$ , рассчитанные в нормах  $C$  и  $L_2$  по формулам

$$\varepsilon_1 = \max_{t \in [t_0, t_n]} |p_1(t) - p_2(t)|; \quad \varepsilon_2 = \sqrt{\frac{1}{t_n - t_0} \int_{t_0}^{t_n} (p_1(t) - p_2(t))^2 dt}.$$

На рис. 3 представлены для сравнения графики интерполяции данных задачи 1, приведенных в табл. 1, кубическим сплайном класса  $C^2$  и сплайнами класса  $C^1$ . Визуально ближе всего к сплайну класса  $C^2$  расположен сплайн, построенный трехэтапным методом. Это подтверждается значениями погрешности из табл. 2.

Данные задачи 2 представлены в табл. 3. Особенность задачи состоит в том, что все точки  $(x_i, y_i)$  принадлежат области комонотонности  $\mathcal{M}$ , но при этом точка для отрезка [28, 33] не принадлежит подобластям  $\mathcal{J}_1$ ,  $\mathcal{J}_2$ . Метод  $F-C$  корректирует эту точку, тогда как трехэтапный и двухходовый алгоритмы оставляют все точки без изменений, что приводит к точному совпадению сплайнов  $M-L$  и  $E-J-L$ . На рис. 4 приведены результаты расчетов задачи 2 разными методами. Вне окрестности точки минимума функциональной последовательности  $f(t_i)$  сплайны  $C^2$  и  $M-L$  совпадают. График сплайна  $E-J-L$  на рисунке не показан. Значения величин  $\varepsilon_1$  и  $\varepsilon_2$  представлены в табл. 4.

Интерполяция сплайнами данных задачи 3 (табл. 5) представлена на рис. 5. Сплайны  $E-J-L$  и  $F-C$  оказались очень близки, отличие заметно лишь на одном отрезке. Из табл. 6 видно, что, как и в предыдущих задачах, наиболее близким к кубическому сплайну класса  $C^2$  является сплайн, рассчитанный с помощью трехэтапного метода.

Таким образом, численные исследования показали, что среди всех опробованных выше методов трехэтапный алгоритм строит сплайн, который лучше других (в специфической задаче 2 не хуже) приближен к кубическому сплайну класса  $C^2$  при оценке отклонения в нормах  $C$  и  $L_2$ . Если все точки  $(d_{i-1}/\delta_i, d_i/\delta_i)$  принадлежат множеству комонотонности  $\mathcal{M}$ , как в задаче 2, то сплайны

Таблица 1

Данные задачи 1

$t$	5,5	5,8	6,2	6,4	7,7	8
$f$	7,2	7,6	3	4	6	10

Таблица 2

Отклонения сплайнов класса  $C^1$  от кубического сплайна класса  $C^2$  в задаче 1

Величина	F-C( $\mathcal{J}_1$ )	F-C( $\mathcal{J}_2$ )	E-J-L	M-L
$\varepsilon_1$	1,83	1,88	2,04	1,63
$\varepsilon_2$	0,82	0,84	0,95	0,73

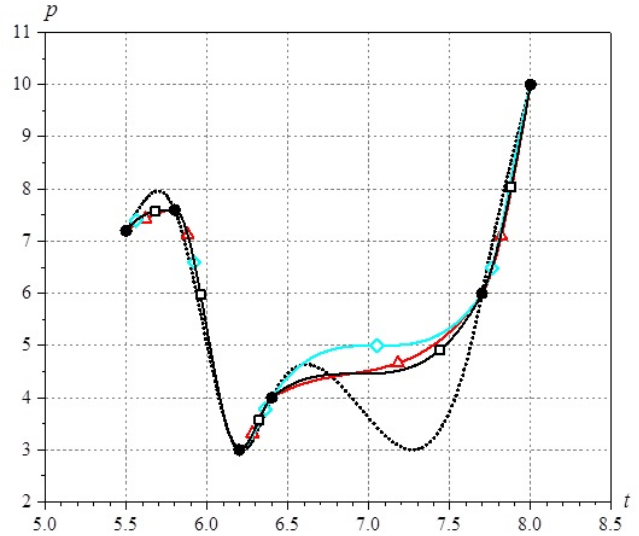


Рис. 3. Интерполяция сплайнами данных задачи 1:  $\bullet - (t_i, f_i)$ ;  $\cdots - C^2$ ;  $\triangle - F-C$ ;  $\diamond - E-J-L$ ;  $\square - M-L$

Таблица 3

Данные задачи 2

$t$	13	17	23	28	33	36
$f$	6,8	2,5	0	3,5	6	11,3

Таблица 4

Отклонения сплайнов класса  $C^1$  от кубического сплайна класса  $C^2$  в задаче 2

Величина	F-C( $\mathcal{J}_1$ )	F-C( $\mathcal{J}_2$ )	E-J-L	M-L
$\varepsilon_1$	0,37	0,37	0,37	0,37
$\varepsilon_2$	0,18	0,18	0,16	0,16

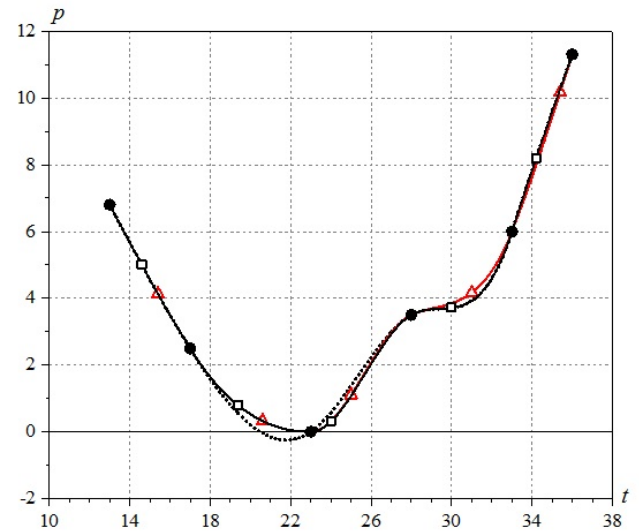


Рис. 4. Интерполяция сплайнами данных задачи 2:  $\bullet - (t_i, f_i)$ ;  $\cdots - C^2$ ;  $\triangle - F-C$ ;  $\square - M-L$

E-J-L и M-L совпадают. При этом они могут отличаться от сплайна класса  $C^2$  лишь на отрезках, примыкающих к точкам экстремумов функциональной последовательности  $f(t_i)$ . Заметим, что метод F-C может, вообще говоря, преобразовать локально монотонный интерполянт класса  $C^2$  не только вблизи экстремумов.

Свойство, согласно которому предлагаемый интерполянт будет немного отличаться от классического сплайна класса  $C^2$ , монотонного на участках монотонности данных (вне окрестностей экстремумов), в некоторых случаях может быть полезным. Дело в том, что для предлагаемого сплайна точки локальных экстремумов совпадают с такими же точками табулируемой функции. И при неотрицательности значений  $f(t_i)$  гарантируется неотрицательность интерполянта (см. рис. 4), тогда как классический сплайн может, вообще говоря, выходить в область отрицательных значений, что не всегда является приемлемым.

Таблица 5

Данные задачи 3

$t$	0	0,06	0,2	0,25	0,34	0,4	0,42
$f$	0	13	14,5	17	15	10,8	3,8

Таблица 6

Отклонения сплайнов класса  $C^1$  от кубического сплайна класса  $C^2$  в задаче 3

Величина	F—C( $\mathcal{J}_1$ )	E—J—L	M—L
$\varepsilon_1$	2,15	2,15	1,81
$\varepsilon_2$	0,97	1,03	0,89

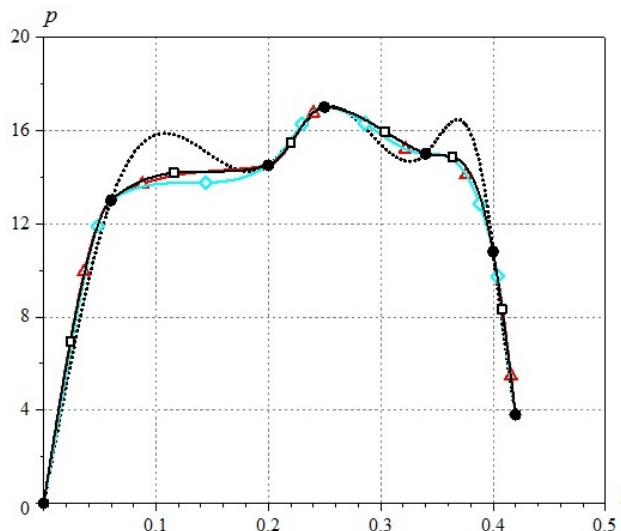


Рис. 5. Интерполяция сплайнами данных задачи 3:  
 • —  $(t_i, f_i)$ ; ····· —  $C^2$ ; —△— — F—C; —◇— — E—J—L;  
 —□— — M—L

### Заключение

В работе представлен трехэтапный метод построения локально комонотонного кубического сплайна класса  $C^1$ . В качестве исходного берется кубический сплайн класса  $C^2$ , для расчета которого используется эффективный метод прогонки.

Очевидно, трехэтапный алгоритм по сравнению с методом F—C является более трудоемким. Тем не менее интерполянт рассчитывается безытерационным методом с использованием элементарных функций. Авторы не выявили увеличения времени счета при сравнении с расширенным двухэтапным алгоритмом, который относится к очень точным.

Достоинством метода является то, что расчет сплайна на первом этапе не зависит от направления обработки интервалов. При этом существует вероятность построения локально комонотонного сплайна уже на первом этапе. В этом случае вычислительные затраты метода на следующих этапах ничтожно малы.

Если данные удастся интерполировать классическим кубическим сплайном класса  $C^2$ , обладающим свойством локальной комонотонности, то трехэтапный сплайн с ним будет совпадать.

Этапы 2 и 3 в предлагаемом методе можно поменять местами, но такой шаг приведет к другому сплайну.

Для численного исследования точности трехэтапного сплайна был привлечен также набор тестов из работ [1] и [2]. По результатам тестирования можно сделать общий вывод о том, что все рассматриваемые сплайны класса  $C^1$  практически совпадают, но для большинства задач предложенный сплайн все же чуть меньше других отклоняется от сплайна класса  $C^2$ .

### Список литературы

1. Квасов Б. И. Монотонная и выпуклая интерполяция весовыми кубическими сплайнами // Журнал вычисл. мат. и мат. физ. 2013. Т. 53, № 10. С. 1610—1621.  
 Kvasov B. I. Monotonnaya i vypuklaya interpolyatsiya vesovymi kubicheskimi splaynami // Zhurnal vychisl. mat. i mat. fiz. 2013. T. 53, № 10. S. 1610—1621.
2. Fritsch F. N., Carlson R. E. Monotone piecewise cubic interpolation // SIAM J. Numer. Anal. 1980. Vol. 17, No 2. P. 238—246.
3. Волков Ю. С. О монотонной интерполяции кубическими сплайнами // Вычислительные технологии. 2001. Т. 6, № 6. С. 14—24.

- Volkov Yu. S.* О монотонной интерполяции кубическими сплайнами // *Vychislitelnye tekhnologii*. 2001. Т. 6, № 6. С. 14–24.
4. *Богданов В. В.* Достаточные условия комонотонной интерполяции кубическими сплайнами класса  $C^2$  // *Математические труды*. 2011. Т. 14, № 2. С. 3–13.  
*Bogdanov V. V.* Dostatochnye usloviya komonotonnoy interpolatsii kubicheskimi splaynami klassa  $S_2$  // *Matematicheskie trudy*. 2011. Т. 14, № 2. С. 3–13.
5. *Fritsch F. N., Butland J.* A Method for constructing local monotone piecewise cubic interpolants // *SIAM J. Stat. Comput.* 1984. Vol. 5, No 2. P. 300–304.
6. *Eisenstat S. C., Jackson K. R., Lewis J. W.* The order of monotone piecewise cubic interpolation // *SIAM J. Numer. Anal.* 1985. Vol. 22, No 6. P. 1220–1237.
7. *Корн Г., Корн Т.* Справочник по математике (для научных работников и инженеров). Определения, теоремы, формулы. 6-е изд. С-Пб.: Лань, 2003.  
*Korn G., Korn T.* Spravochnik po matematike (dlya nauchnykh rabotnikov i inzhenerov). Opredeleniya, teoremy, formuly. 6-e izdanie. S-Pb.: Lan, 2003.

Статья поступила в редакцию 01.10.20.

ABOUT ONE LOCALLY COMONOTONE CUBIC  $C^1$ -CLASS SPLINE / S. V. Mzhachikh, Yu. N. Lapshina (FSUE "RFNC-VNIIEF", Sarov, N. Novgorod region).

In some studies it is required, from the standpoint of physics, that the data interpolating curve is monotone in each data monotonicity interval. The use of the classical cubic  $C^2$ -class spline is not always possible for such problems. However, this problem is solvable and there are different ways to solve it. The paper presents a cubic  $C^1$ -class spline for solving the monotone interpolation problem. This spline coincides with the classical cubic  $C^2$ -class spline in the monotone behavior sections of the functional sequence under the condition that the classical spline is monotone in these sections. The only difference is observed near local extrema. Numerical results of the accuracy examination are presented, with the new interpolant being compared to other splines of some popular algorithms.

*Keywords:* cubic spline, monotone interpolation, local co-monotonicity, the Fritsch-Carlson method.

---