

СООБЩЕНИЯ
ОБЪЕДИНЕННОГО
ИНСТИТУТА
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

Дубна

99-168

P10-99-168

Н.Д.Дикусар

КУСОЧНО-КУБИЧЕСКОЕ ПРИБЛИЖЕНИЕ
И СГЛАЖИВАНИЕ КРИВЫХ В РЕЖИМЕ
АДАПТАЦИИ

1999

1. Введение

Создание простых, надежных и экономичных сглаживающих алгоритмов является актуальной задачей для широкого спектра исследований и практических приложений, в частности при обработке экспериментальных данных и численном решении прикладных задач в различных областях науки и техники [1,9,10,11,12].

В прикладных задачах, как правило, функции часто задаются не формулой, а с помощью какого-либо вычислительного процесса различной сложности. Например, функции могут являться результатом численного интегрирования дифференциального уравнения на заданном отрезке или вычисляться по сложной итерационной схеме и т.п. В таких случаях их называют *вычислимыми функциями*. Значения таких функций известны с заданной точностью, однако их явный вид заранее не известен, так же, как и их общие свойства. Все операции над такими функциями могут выполняться только посредством вычисления значений самой функции. Более сложные ситуации возникают при обработке эмпирических данных, получаемых в измерениях при наличии посторонних шумов, например, в системах автоматической обработки контурных изображений, в системах адаптивного управления динамическими объектами, при цифровой обработке сигналов и т.п. [1, 9 -12]. Как правило, анализ данных в таких системах выполняется в режиме реального времени, для которого существуют жесткие ограничения и противоречивые требования к алгоритмам обработки. Такие алгоритмы должны обеспечивать заданную точность и высокую скорость вычисления параметров при ограниченных ресурсах памяти и быть устойчивыми к фону и ошибкам. Эти требования существенно ограничивают применение классических методов адаптивной фильтрации, например, рекурсивного алгоритма наименьших квадратов (РНК) высокого порядка или оптимального калмановского оценивания из-за их вычислительной сложности и неустойчивости [1].

В работе предлагается новый подход к практическому решению задачи сглаживания экспериментальных данных и локально-оптимального приближения функций. Этот подход основан на использовании 4-точечных или дискретных проективных преобразований (DPT) [2] и РНК первого порядка. В качестве локальной аппроксиманты предложена модель 3-точечного кубического сплайна (TPS-модель) [2] с одним свободным параметр θ . Остальные три параметра являются фиксированными и совпадают с ординатами трех точек кривой $\mathbf{R}_0^T = [R_x, R_L, R_0]$, которые заранее известны, в общем слу-

чае - с некоторой погрешностью. Такая конструкция модели является достаточно гибкой при описании локальной формы кривой в режиме адаптации.

Предложенный на основе TPS-модели метод локального приближения и сглаживания данных, ориентирован, прежде всего, на обработку экспериментальных данных в режиме реального времени. Он также может быть использован для работы с вычислимыми функциями и с функциями, заданными формулами, например, для нахождения узлов глобального кубического сплайна.

Метод работает в два этапа. На первом этапе используются приближенные значения фиксированных параметров $\tilde{\mathbf{R}}_0$ для вычисления оценки $\hat{\theta}$ и числа точек n_θ ("объем выборки"). На основе ДРТ и РНК 1-го порядка строится итерационная процедура для вычисления $\hat{\theta}$, типа известной в теории стохастической аппроксимации, процедуры Роббинса - Монро [3,7], в которой коэффициент усиления (влияния) $\gamma(n)$ изменяется не хуже чем $1/n^3$ (n - индекс итерации), что значительно ускоряет процесс адаптации. На втором этапе полученные значения $\hat{\theta}$ и n_θ используются для коррекции фиксированных параметров $\hat{\mathbf{R}}_0$ по методу наименьших квадратов (МНК). В этом случае вычислительная схема использует в два раза меньше операций по сравнению с алгоритмом РНК 3-го порядка [1].

Возможности метода показаны на примере создания алгоритма для кусочно-кубического приближения (сглаживания) кривой $f(x) \in C$, $x \in [a, b]$, заданной последовательностью N дискретных точек $\{x_k, f_k\}$, $k = 1, 2, \dots, N$, ($N \gg 4$) в виде суммы M , ($M \geq 1$) локальных кубических сплайнов $S_j(x; \hat{\Theta})$

$$f(x) \approx \sum_{j=1}^M \alpha_j S_j(x; \hat{\Theta}), \quad \alpha_j = \begin{cases} 1, & x \in [x_{l_j}, x_{r_j}] \\ 0, & x \notin [x_{l_j}, x_{r_j}] \end{cases}, \quad [x_{l_j}, x_{r_j}] \subseteq [a, b], \quad M \geq 1, \quad (1)$$

при условии, что погрешность приближения не превышает заданную величину T_f (порог точности)

$$\max_{x \in [x_{l_j}, x_{r_j}]} |f - S_j| < T_f, \quad j = 1, 2, \dots, M, \quad (2)$$

где $\hat{\Theta}$ - набор параметров j -го сплайна, включающий его левый и правый узлы - x_{l_j} и x_{r_j} . Быстрое и надежное решение задачи (1)-(2) актуально при создании систем обработки данных, например, в задачах по обнаружению треков в экспериментах по физике высоких энергий [11], при распознавании контуров треков α -частиц [12] в условиях высокого фона и ошибок оцифровки и др.

Как известно [4], оптимальный выбор узлов при представлении $f(x)$ интерполяционным сплайном является весьма трудной задачей, а в случае применения метода наименьших квадратов выбор подходящего набора точек разбиения для оценки аппроксимирующего сплайна часто можно определять только методом проб и ошибок. В этом случае локальное сглаживание функций, искаженных случайными помехами $e(x)$, существенно усложняется при

увеличении дисперсии ошибки σ_c^2 . Разложение (1)-(2) функции $f(x)$, заданной аналитически (формулой), может представлять интерес также с вычислительной точки зрения, например, в задачах поиска локального экстремума или начального приближения корней функции.

Табличное задание функции $\{\tilde{f}_k\}$ характерно для систем обработки экспериментальных данных, в которых задача (1)-(2) может быть использована как быстрая процедура для определения параметров или признаков по результатам измерений, а также как программный инструмент для сжатия данных.

При заданном пороге T_f "нарезка" $f(x)$ в виде локальных кубических сплайнов дает возможность быстро получать информацию о локальных особенностях кривой и тем самым сокращать объем вычислений. Например, в задачах распознавания контуров или треков определение характеристик объекта (длина кривой, площадь и т.п.) можно находить, не используя строгую гладкость кривой и ее производных в точках сопряжения сплайнов, хотя алгоритм позволяет корректировать гладкость сопряжения S_j путем подбора параметров весовых функций, шага квантования и порога точности.

Предложенный метод нахождения оценок параметров локальных кубических сплайнов может быть использован при создании адаптивных сглаживающих алгоритмов (фильтров), простых в вычислениях и экономичных в использовании оперативной памяти, ориентированных на обработку данных в режиме их поступления, что дает преимущество по сравнению с традиционными подходами, использующими заранее известные выборки данных.

2. Выбор модели локальной аппроксиманты

В этом разделе рассматриваются структура и свойства 4-точечных преобразований (DPT) и 3-точечной модели локального кубического сплайна (TPS).

2.1. DPT, или 4-точечное преобразование

На отрезке $[a, b]$ зададим параметры x_0 (базисная точка) и λ, L (полюсы), $\lambda \neq L$ и зафиксируем на кривой $\tilde{f}(x) = f(x) + e(x)$, $f(x) \in C$, $x \in [a, b]$, ($e(x)$ - случайная ошибка) три точки или репер

$$\mathcal{R} : \{(x_0, \tilde{R}_0); (x_0 + \lambda, \tilde{R}_\lambda); (x_0 + L, \tilde{R}_L)\}, \text{ где } \tilde{R}_\cdot \equiv \tilde{f}(x_\cdot).$$

Для любой неполюсной точки $x_k \in [a, b]$, $k = 1, 2, \dots$ найдем $\tau_k = x_k - x_0$, $\tilde{\phi}_k = \tilde{f}_k - \tilde{f}_0$, $\tilde{\theta}_\lambda \equiv \tilde{\phi}_\lambda = \tilde{R}_\lambda - \tilde{R}_0$, $\tilde{\theta}_L \equiv \tilde{\phi}_L = \tilde{R}_L - \tilde{R}_0$, вычислим весовые векторы $\mathbf{P}_k^T = [p_{1k}, p_{2k}, p_{3k}]$, $\mathbf{D}_k^T = [d_{1k}, d_{2k}, d_{3k}]$ и составим векторы наблюдений $\tilde{\mathbf{Y}}_k^T = [\tilde{\theta}_\lambda, \tilde{\theta}_L, \tilde{\phi}_k]$ и $\tilde{\mathbf{Z}}_k^T = [\tilde{\theta}_\lambda, \tilde{\theta}_L, \tilde{\phi}_k^*]$.

Тогда прямое 4-точечное преобразование функции $\tilde{f}(x)$ в точке x_k для заданного репера \mathfrak{R} , согласно [2], определяется в виде свертки вектора наблюдений \tilde{Y}_k с весовым вектором \mathbf{P}_k

$$\tilde{\phi}_k^a \equiv \tilde{\phi}^a(x_k, \mathfrak{R}) = (\tilde{Y}_k, \mathbf{P}_k), \quad k=1,2,\dots, \quad (3)$$

а свертка вектора \tilde{Z}_k с весовым вектором \mathbf{D}_k определяет обратное для (3) 4-точечное преобразование

$$\tilde{\phi}_k^b \equiv \tilde{\phi}^b(x_k, \mathfrak{R}) = (\tilde{Z}_k, \mathbf{D}_k), \quad k=1,2,\dots, \quad (4)$$

Координаты весовых векторов \mathbf{P}_k и \mathbf{D}_k задаются функциями $p_{ik} = p_i(\tau_k; \lambda, L)$ и $d_{ik} = d_i(\tau_k; \lambda, L)$, ($i=1,2,3$) (рис.1), которые определяются двойным отношением $\frac{13}{24} : \frac{33}{14}$ четырех точек оси x , взятых на алгебраических расстояниях $0, \lambda, L$ и τ_k от базисной точки x_0 . В приведенной записи сложного отношения цифры $1, 2, 3, 4$ указывают порядковый номер точки в четверке, например, число 13 означает расстояние $x_1 - x_3$ и т.д. [2]. На основе "производящей" функции этого отношения

$$p_3(\tau_k; \lambda, L) = \frac{\lambda L}{(\tau_k - \lambda)(\tau_k - L)}, \quad \lambda \neq L \neq \tau_k, \lambda, L \neq 0 \quad (5)$$

легко найти другие весовые функции. Например, p_{1k} и p_{2k} получаются из (5) простыми перестановками $\lambda \leftrightarrow \tau_k$ и $L \leftrightarrow \tau_k$ в p_{3k} , а в силу $p_{3k} \neq 0$ и свойства нормировки p_i в текущей точке τ_k , функции d_{ik} выражаются через функции p_{ik} в виде

$$d_{ik} = \frac{(-p_{ik})^j}{p_{3k}}, \quad \text{где } j = (3i - i^2)/2, \quad i=1,2,3.$$

При дискретном представлении кривой $f(x)$, в силу масштабной инвариантности отношения (5), функции $p_{ik} = p_i(k; k_\lambda, k_L)$ и $d_{ik} = d_i(k; k_\lambda, k_L)$ становятся функциями индекса k . Значения $k_\lambda = \lambda/h$ и $k_L = L/h$ выбираются из множества индексов $\{k\}$ при $k_\lambda \neq k_L$, где h - шаг сетки. В дальнейшем будем их обозначать через μ и m , соответственно.

2.2. Модель 3-точечного кубического сплайна (TPS-модель)

В [2] была предложена формула для приближения функции $f(x) \in C$ системой N_α ($N_\alpha \geq 3$) моносплайнов k -го порядка $S_k(\tau; \lambda, L)$ и квадратичной

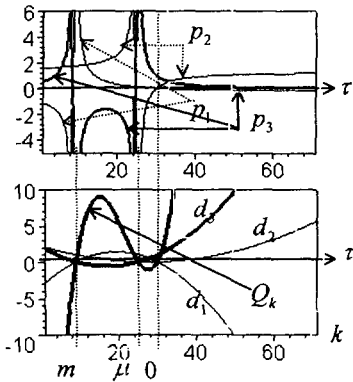


Рис.1. Пример графиков весовых функций

параболы $\Pi(\tau; \mathfrak{R})$, фиксированной "реперными координатами" на отрезке $[x_\lambda, x_L]$ в виде

$$f(x) \approx \Pi(\tau, \mathfrak{R}) + \sum_{k=3}^{N_k} \alpha_k(\lambda, L) S_k(\tau; \lambda, L), \quad (6)$$

где $S(\tau; \lambda, L)$ имеют нули в точках λ, L и 0 , $\alpha_k(\lambda, L)$ - неизвестные параметры, а $\tau = x - x_0$. Такая конструкция моносплайнов дает равномерный характер приближения $f(x)$ на отрезке, а базис $\{S_k\}$ обеспечивает устойчивость вычислений при $\tau \rightarrow 0$. При $N = 3$ формула (6) дает нам модель локального трехточечного кубического сплайна в виде

$$S(\tau; \Theta) = S_3(\tau; \mathfrak{R}) = (\mathbf{R}_0, \mathbf{D}) + \theta Q(\tau; \lambda, L), \quad (7)$$

где θ - неизвестный (свободный) параметр, $(\mathbf{R}_0, \mathbf{D}) = \Pi(\tau; \mathfrak{R})$ - квадратичная парабола, проходящая через реперные точки $\mathbf{R}'_0 = [R_\lambda, R_L, R_0]$, $\mathbf{D}' = [d_1, d_2, d_3]$, а $Q(\tau; \lambda, L) = \tau(\tau - \lambda)(\tau - L)$ - "зануляющая" кубическая парабола (рис.1, 2).

При известном репере \mathfrak{R} модель (7) позволяет свести процедуру локального приближения $f(x)$ к поиску оценки только одного параметра θ . Например, если $f(x)$ задана аналитически, то параметр θ определяется точной формулой [2]:

$$\theta = \frac{1}{H^2} [f'(x_\lambda) + f'(x_L) - \frac{2}{H}(f(x_L) - f(x_\lambda))], \quad H = L - \lambda. \quad (8)$$

Из (7) и (8) следует, что кубическое приближение $f(x)$, $x \in [x_\lambda, x_L]$ на основе TPS-модели использует $f(x)$ в точках x_0, x_λ, x_L и $f'(x)$ в x_λ, x_L . Для приближения функции многочленом третьей степени по формуле Тейлора используются значения $f(x), f'(x), f''(x)$ и $f'''(x)$ в точке x_0 . Для $x_\lambda < x_0 < x_L$ формула (6) дает равномерный характер погрешности на отрезке $[x_\lambda, x_L]$. Это свойство TPS-модели весьма полезно и при сглаживании вычислимой функции, заданной в дискретном виде $\tilde{f}_k = f(x_k) + e(x_k)$, $x_k = x_0 + kh$, ($k = 0, \pm 1, \pm 2, \dots, \pm n; |n| \gg 4$) и искаженной случайными ошибками.

Как было показано в [2], применение модели (7) и преобразования (3) для сглаживания $\{\tilde{f}_k\}$ по МНК, дает следующую оценку параметра θ :

$$\hat{\theta} = (\lambda L \sum_{k=1}^n \tau_k^2)^{-1} \sum_{k=1}^n \tau_k \tilde{\phi}_k^a. \quad (9)$$

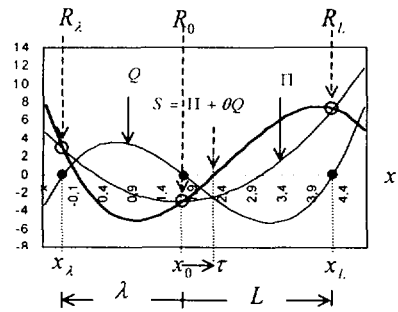


Рис.2. TPS-модель

При этом, чтобы избежать "грубых выбросов" точек, преобразованных по формуле (3), значения τ_k необходимо выбирать вне "шумовых зон", определяемых порогом T_v : $|\tau_k - \lambda| < T_v$ и $|\tau_k - L| < T_v$. По сравнению со стандартной кубической моделью с четырьмя коэффициентами, TPS-модель (7) является более гибкой и позволяет значительно упростить процесс нахождения оценок его параметров.

2.3. Устойчивость DPT к ошибкам

Важным свойством преобразований (3) является их устойчивость к аддитивному шуму [2], что было использовано при создании адаптивных проективных фильтров для обнаружения треков (APF) [5], а также при разработке новых алгоритмов обработки [6, 8].

Операция (3) обладает рядом свойств. Например, если $f(x)$ - полином степени n , то $f^\circ(x; \mathfrak{R})$ будет полиномом степени $n - 2$, квадратичная парабола и прямая линия преобразуются в константу, а константа преобразуется в себя. Применение DPT-операции к TPS-модели дает

$$S^\circ(\tau; \mathfrak{R}, \theta) = \Pi^\circ(\tau; \mathfrak{R}) + \theta Q^\circ(\tau; \lambda, L) = S_0 + \theta \lambda L \tau. \quad (10)$$

Как видно из (10), кубическая дуга преобразуется в линейный отрезок с угловым коэффициентом $\theta \lambda L$ и свободным членом S_0 .

Если точки кубической параболы искажены случайными ошибками $e(x)$, тогда $\tilde{\mathbf{S}} = \mathbf{S} + \mathbf{E}$, где $\mathbf{E}^T = [e_\lambda, e_l, e_x]$ - вектор ошибок, а $\tilde{\mathbf{S}}^T = [\tilde{S}_\lambda, \tilde{S}_l, \tilde{S}_x]$. В этом случае преобразование точки (x, \tilde{S}_x) при заданном репере \mathfrak{R} происходит по правилу

$$\tilde{S}^\circ(x; \mathfrak{R}) = (\tilde{\mathbf{S}}, \mathbf{P}) = (\mathbf{S}, \mathbf{P}) + (\mathbf{E}, \mathbf{P}) = \tilde{S}_0 + \theta \lambda L \tau + e_x^\circ,$$

т.е. преобразование ошибки $e(x)$ происходит по схеме свертки вектора ошибок с вектором весовых функций

$$e_x^\circ = (\mathbf{E}, \mathbf{P}) = \varepsilon_x. \quad (11)$$

Это уравнение ошибки показывает, что операция DPT подавляет случайную погрешность квадратичным образом, так как $e(x)$ преобразуется через знаменатель функций p_k (5). При этом если точки $\{e_\lambda, e_l, e_0, e_x\}$ лежат на прямой или на квадратичной параболе, то для вектора $\mathbf{E}_0^T = [e_\lambda - e_0, e_l - e_0, e_x - e_0]$ получим

$$(e_x - e_0)^\circ = (\mathbf{E}_0, \mathbf{P}) = 0, \quad (12)$$

что свидетельствует об устойчивости преобразования к такой систематике.

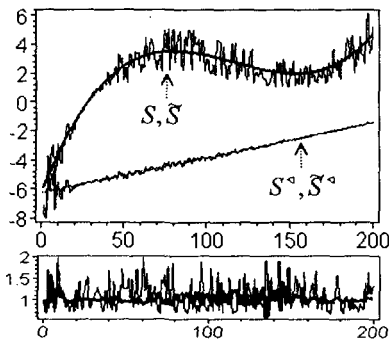


Рис. 3. Графики S , \tilde{S} , S° , \tilde{S}° , S_k / \tilde{S}_k и $S_k^\circ / \tilde{S}_k^\circ$ ("жирный") ($\mu = -53, m = -57$)

Пример применения 4-точечного преобразования к зашумленной кубической кривой показан на рис.3. В нижней части рисунка приведены графики отношений S_k/\tilde{S}_k и S_k^a/\tilde{S}_k^a , показывающие эффект уменьшения такого отношения для преобразованных данных.

Отмеченные свойства позволяют применить простую линейную регрессию для сглаживания преобразованных точек $\{\tilde{S}_k^a\}$ вместо исходных $\{\tilde{S}_k\}$. После этого, с помощью обратного преобразования (4), получим сглаженную исходную кривую. Такой подход дает ряд преимуществ по сравнению с традиционной схемой сглаживания на основе кубической модели. Например, так как ординаты реперных точек принимаются в качестве оценок параметров $\hat{\mathbf{R}}_0$, то задача нахождения оптимального кубического сплайна для локального приближения (сглаживания) кривой сводится к определению оценки только параметра θ и двух величин x_L, x_R - граничных узлов. Основная трудность при этом состоит в поиске участка кривой, адекватного по форме TPS-модели. Несоответствие модели кривой вызывает отклонение преобразованных точек от прямой линии, что увеличивает ошибку в оценке $\hat{\theta}$. При выборе репера \mathcal{R} необходимо учитывать величину шага h и дисперсию ошибки σ_e^2 . Например, в случае сглаживания усреднение "реперных координат" по трем или пяти соседним точкам заметно улучшает процесс адаптации. Влияние погрешности вектора $\tilde{\mathbf{R}}_0$ на точность подгонки и на длину промежутка слежения $[x_L, x_R]$ можно регулировать также подбором параметров λ и L .

Конкретный вид кубической дуги $S(x; \hat{\Theta})$ существенно зависит также от выбора значений параметров настройки алгоритма h, T_f, T_v , в зависимости от величины дисперсии σ_e и от сложности самой кривой $f(x)$. В частности, с помощью этих параметров можно регулировать "плавность" сопряжения сплайнов и точность оценки $\hat{f}(x)$ при решении задачи (1) - (2).

Принимая ординаты реперных точек в качестве приближенных значений для фиксированных параметров $\tilde{\mathbf{R}}_0$, поиск локального оптимального кубического сплайна $S_j(x; \hat{\Theta})$ можно осуществить в два этапа:

- 1) вычислить оценку $\hat{\theta}$, параметр n_θ и найти узлы x_L, x_R ;
- 2) на основе полученных данных получить оценки $\hat{\mathbf{R}}_0$.

Величины $\hat{\mathbf{R}}_0, \hat{\theta}, n_\theta, x_L, x_R$ используются при формировании набора $\hat{\Theta}$ для $S_j(x; \hat{\Theta})$ в задаче (1)-(2).

3. Локально-оптимальное кубическое сглаживание кривых

В этом разделе на базе TPS-модели, DPT и РНК строится итерационная процедура типа известной в методе стохастической аппроксимации процедуры

Роббинса-Монро [3,7] вычисления оценки свободного параметра $\hat{\theta}$, которая затем используется для коррекции фиксированных параметров $\tilde{\mathbf{R}}_0$ с применением стандартной процедуры МНК. Построен простой оптимально-локальный сглаживающий (аппроксимирующий) алгоритм третьего порядка (LOCUS) с поэтапным вычислением оценок параметров TPS-модели.

3.1. Итерационная процедура для вычисления оценки $\hat{\theta}$

Рассмотрим случай, когда значения функции (отсчеты) $\tilde{f}_k = f(x_k) + e_k$ заданы на сетке с шагом h , $\{x_k = x_0 + kh\}$, $k = 0, 1, 2, \dots, N$, $x_k \in [a, b]$, а ошибки e_k распределены нормально с нулевым математическим ожиданием и дисперсией σ_e^2 . При $\sigma_e^2 = 0$, $\tilde{f}_k \equiv f(x_k)$, т. е. функция вычисляется аналитически.

Часто в современных экспериментах поток поступающих данных образует временную последовательность, и всегда возникает необходимость получать оценки параметров в любой момент времени по информации, накопленной до этого момента. В этом случае используется рекуррентное вычисление оценок наименьших квадратов, так как объем выборки (параметр n) заранее неизвестен и применение стандартной процедуры МНК становится малоэффективным.

Оценка $\hat{\theta}$ в формуле (9) получена из условия минимума суммы квадратов отклонений $\varepsilon_k^a = (\tilde{f}_k - S_k(\lambda, L; \hat{\theta}))^a$ при известном числе точек n :

$$\sum_{k=1}^n (\varepsilon_k^a)^2 \rightarrow \min_{\theta}.$$

Используя (9) для вычисления оценки $\hat{\theta}$ сначала по $n-1$, а затем по n точкам, получим рекуррентную формулу для вычисления $\hat{\theta}_n$ через $\hat{\theta}_{n-1}$ в следующем виде:

$$\hat{\theta}_n = \hat{\theta}_{n-1} + \gamma_n [\tilde{\phi}_n^a - \hat{\theta}_{n-1} \lambda L \tau_n],$$

где

$$\gamma_n \equiv \gamma(\tau_n; \lambda, L) = \tau_n (\lambda L \sum_{k=1}^n \tau_k^2)^{-1},$$

а выражение в квадратных скобках равно 4-точечному преобразованию отклонения ε_n в точке τ_n :

$$\varepsilon_n^a = [\tilde{\phi}_n - S_n(\lambda, L; \hat{\theta})]^a = \tilde{\phi}_n^a - S_n^a.$$

Отсюда, при $\tau_n = nh$, $\lambda = \mu h$, $L = mh$ и с учетом равенства

$$\sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6}$$

получим

$$\hat{\theta}_n = \hat{\theta}_{n-1} + \gamma_n [\tilde{\phi}_n^a h^{-3} - \hat{\theta}_{n-1} \mu mn], \quad n = 1, 2, \dots, \quad (13)$$

где

$$\gamma_n = \frac{6}{\mu n(n+1)(2n+1)}. \quad (14)$$

Начальное значение $\hat{\theta}_0$ находим по значению образа функции $\tilde{\phi}_n^a$ в точке x_{-1} ($n = -1$), ближайшей к базису x_0 :

$$\hat{\theta}_0 = -\tilde{\phi}_{-1}^a / \mu m h^3. \quad (15)$$

При заданном пороге точности T_f критерием окончания итераций (13) служит неравенство $|r_n| > T_f$, где

$$r_n = \tilde{f}_n - S_n = \tilde{f}_n - (\tilde{\mathbf{R}}_0, \mathbf{D}_n) - \hat{\theta}_n Q_n = \tilde{\phi}_n - (\hat{\mathbf{V}}_n, \mathbf{W}_n), \quad (15a)$$

$\hat{\mathbf{V}}_n^T = [\tilde{\theta}_2, \tilde{\theta}_l, \hat{\theta}_n]$ - вектор параметров и $\mathbf{W}_n^T = [d_{1n}, d_{2n}, Q_n]$ - весовой вектор.

Для исключения фоновых точек с ошибками $|e_x| > 3\sigma_e$ в процедуру вычисления оценки (13) можно включить контроль величины коррекции свободного параметра $\Delta \hat{\theta}_n = \hat{\theta}_n - \hat{\theta}_{n-1}$, в виде $|\Delta \hat{\theta}_n| < T_\theta$, где T_θ - заданный порог.

Значение $n_\theta \equiv n$, полученное в момент прекращения итерационного процесса, определяет "размер выборки" или число точек, которые были использованы для вычисления оценки $\hat{\theta} \equiv \hat{\theta}_n$. После этого концевые точки (узлы) локального сплайна определяются в виде $x_l = x_0 + h$ и $x_R = x_0 + n_\theta h$.

Формула (13) представляет хорошо известную в методе стохастической аппроксимации процедуру Роббинса-Монро [3, 7], связанную с общим классом рекурсивных алгоритмов, используемых для решения уравнений вида

$$g(\Omega) = E\{G(y, \Omega)\} = 0,$$

где G - известная функция входного сигнала y , Ω - вектор неизвестных параметров, а $E\{\cdot\}$ - символ математического ожидания. Такие алгоритмы применяются, например, для решения задач отыскания корней и экстремумов функции регрессии [3].

В процедуре Роббинса-Монро для корректировки Ω на основе последовательности наблюдений y_k используется соотношение

$$\hat{\Omega}_k = \hat{\Omega}_{k-1} + \gamma_k G[y_k, \hat{\Omega}_{k-1}], \quad k = 1, 2, \dots, \quad (16)$$

где γ_k - специальным образом подобранная последовательность, которая должна удовлетворять следующим условиям:

$$\lim_{k \rightarrow \infty} \gamma_k = 0, \quad \sum_{k=1}^{\infty} \gamma_k = \infty \quad \text{и} \quad \sum_{k=1}^{\infty} \gamma_k^2 < \infty. \quad (17)$$

Второе из этих условий гарантирует достаточное число шагов корректировки, что позволяет близко подойти к искомому решению, тогда как третье условие гарантирует конечность дисперсии накопленного шума. Функция \hat{G} ограничена и является несмещенной оценкой G : $E\{\hat{G}\} = G$, т. е.

G - это функция регрессии для случайного процесса \tilde{G} . При выполнении условий (17) процедура (16) сходится в среднеквадратичном смысле (см. [3]), т. е.

$$\lim_{k \rightarrow \infty} E\{(\Omega_k - \Omega_0)^2\} = 0, \text{ где } \Omega_0 - \text{корень уравнения } G = 0.$$

Методы стохастической аппроксимации (МСА) применяются для решения прикладных задач в области техники, биологии, теории управления, обучения и т.п. Хотя сходимость МСА доказана строго математически, его практическое применение не всегда удовлетворяет решению прикладных задач, так как эта сходимость проявляется при $k \rightarrow \infty$. В практических расчетах необходимо исследовать некоторую окрестность экстремума за небольшое число шагов [10]. Поэтому вопрос выбора коэффициента усиления γ_k , определяющего скорость адаптации и сходимость процедуры (16), является весьма важным. Как известно, на практике чаще всего используется гармоническая последовательность $\{\gamma_k = 1/k^q\}$, $1/2 < q \leq 1$, $k = 1, 2, \dots$, удовлетворяющая условиям (17).

Возвращаясь к рекуррентной формуле (13), находим, что выражение в квадратных скобках соответствует функции $G[y_k, \hat{\Omega}_{k-1}]$ из (16), т. е.

$$G[\tilde{\phi}_n^q, \hat{\theta}_{n-1}] = \tilde{\phi}_n^q h^{-3} - \hat{\theta}_{n-1} \mu m n.$$

Знаменатель последовательности $\{\gamma_n\}$ (14) изменяется квадратично по n и зависит также от параметров μ и m , что позволяет более чем на два порядка увеличивать скорость убывания коэффициента γ_n по сравнению с $1/n$. Для последовательности (14) выполняются только первое и третье условия из (17), т.е.

$$\lim_{n \rightarrow \infty} \gamma_n = 0 \text{ и } \sum_{n=1}^{\infty} \gamma_n^2 = \frac{12}{\mu^2 m^2} (2\pi^2 - 24 \ln 2 - 3) < \infty.$$

Однако, в силу сходимости ряда

$$\sum_{n=1}^{\infty} \gamma_n = \frac{6}{\mu m} (2 \ln 2 - 1) < \infty,$$

второе условие из (17) не выполняется. Но, так как сумма ряда достигает своего предела, начиная с $n \approx 10^9$, то этого вполне достаточно для практического применения метода. Зависимость γ_n от параметров μ и m дает дополнительную возможность влиять не только на скорость сходимости итерационного процесса, но и на подавление случайных помех. Подходящим выбором этих параметров можно добиться скорости сходимости итераций, значительно превышающей $1/n^3$ (рис.4).

Максимальную ошибку вычисления оценки $\hat{\theta}_n$ можно найти с помощью формул (5), (12), (13) и (14) в виде:

$$\hat{\varepsilon}_{\hat{\theta}_n} = \gamma_n e_n^q h^{-3} = \gamma_n (\mathbf{E}_0, \mathbf{P}) h^{-3} < \gamma_n h^{-3} \sum_{i=1}^3 |p_{im}| e_{\max} = \beta_n h^{-3} e_{\max},$$

где $e_{\max} = \max\{e_{\lambda}, -e_0, e_l - e_0, e_n - e_0\}$, а коэффициент $\beta_n = \beta(n; \mu, m)$, с учетом (5), выражается формулой

$$\beta_n = \gamma_n \left[\left| \frac{mn}{(n-\mu)v} \right| + \left| \frac{n\mu}{(n-m)v} \right| + \left| \frac{\mu m}{(n-\mu)(n-m)} \right| \right], \quad v = m - \mu. \quad (18)$$

Из (13) и (18) следует, что при вычислении оценки $\hat{\theta}_n$, для эффективного подавления случайных ошибок входных данных, значения μ и m необходимо выбирать в области отрицательных чисел. На рис.4 приведен график поведения

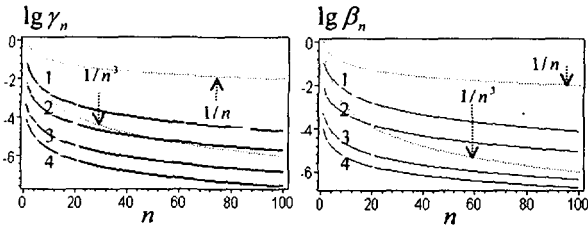


Рис.4. Графики $\lg \gamma_n$ и $\lg \beta_n$ для различных (μ, m) ; (1: $(-3, -5)$; 2: $(-10, -15)$; 3: $(-35, -57)$; 4: $(-100, -110)$)

коэффициента β_n в логарифмическом масштабе для различных значений параметров μ и m , из которого видно, что в начале, на критическом участке слежения, подавление ошибок происходит даже лучше чем $1/n^3$. Для сравнения на рис.4 приведены в том же масштабе графики для $\gamma_n \equiv \gamma(n)$, $1/n$ и $1/n^3$.

3.2. Коррекция фиксированных параметров $\tilde{\mathbf{R}}_0$

При вычислении оценки $\hat{\theta}$, в силу (18), происходит эффективное подавление ошибки, вследствие чего влияние погрешностей в значениях фиксированных параметров (координат вектора $\tilde{\mathbf{R}}_0$) на точность вычисления $\hat{\theta}$ заметно снижается. Кроме того, как отмечалось выше, влияние грубой ошибки в значениях фиксированных параметров можно уменьшить простым усреднением по трем, ближайшим к опорной координате точкам или пятиточечным сглаживанием. Такое усреднение существенно влияет на точность определения оценки $\hat{\theta}$ при большой дисперсии σ_e^2 . При этом, для сохранения плавности при стыковке соседних сегментов, вектор $\tilde{\mathbf{R}}$ (новые полюсы) для следующего сегмента следует формировать по точкам предыдущего, принимая его правый узел за новую базисную точку. Этого можно добиться с помощью параметров μ , m и h . В этом случае два соседних сегмента кривой будут иметь три общие точки. Если погрешности отсутствуют

(случай аппроксимации), тогда фиксированные параметры известны и процедура их коррекции не требуется.

Для коррекции фиксированных параметров используем параболу $\Pi(\tau; \mathfrak{R})$ для сглаживания разности $\hat{\tilde{\varphi}}_k = \tilde{\varphi}_k - \hat{\theta} Q_k$, где $Q_k = h^3 Q(k; \mu, m)$. После сдвига начала координат в точку (x_0, \tilde{f}_0) коэффициент при d_{30} в выражении для $\Pi(\tau; \mathfrak{R})$ исчезнет, вместо $\tilde{\mathbf{R}}_0$ появится вектор $\tilde{\mathbf{C}}^T = [\tilde{\theta}_\lambda, \tilde{\theta}_l]$, а квадратичная модель примет вид

$$U_k = \theta_\lambda d_{1k} + \theta_l d_{2k}.$$

Тогда, используя $\hat{\theta}$ и n_θ , оценку $\hat{\mathbf{C}}$ можно найти с помощью стандартной процедуры метода наименьших квадратов на основе минимизации функционала

$$\chi^2 = \sum_{k=1}^{n_\theta} (\hat{\tilde{\varphi}}_k - U_k)^2 \rightarrow \min_{\theta_\lambda, \theta_l}.$$

МНК-оценка $\hat{\mathbf{C}}^T = [\hat{\theta}_\lambda, \hat{\theta}_l]$ в этом случае запишется в виде

$$\hat{\mathbf{C}}^T = (\mathbf{A}^T \mathbf{A})^{-1} \tilde{\mathbf{B}}^T, \quad (19)$$

где $\mathbf{A}^T \mathbf{A}$ - невырожденная симметричная матрица размерности 2×2 , с $\sum d_{1k}^2$, $\sum d_{2k}^2$ на диагонали, недиагональными элементами $\sum d_{1k} d_{2k}$ и вектором правых частей $\tilde{\mathbf{B}}^T = [\sum \hat{\tilde{\varphi}}_k d_{1k}, \sum \hat{\tilde{\varphi}}_k d_{2k}]$.

При вычислении оценок $\hat{\theta}$, $\hat{\theta}_\lambda$ и $\hat{\theta}_l$ ошибка e_0 в $\tilde{\mathbf{R}}_0$ приравнивалась нулю путем ее вычитания из ошибок во всех других точках, что приводило к некоторому изменению входных погрешностей. Поэтому коррекцию параметра $\tilde{\theta}_{x_0} \equiv \tilde{\mathbf{R}}_0$ получаем обычным усреднением с использованием найденных оценок и n_θ :

$$\hat{\theta}_{x_0} = \frac{1}{n_\theta} \sum_{k=1}^{n_\theta} (\tilde{f}_k - \hat{U}_k - \hat{\theta}_k Q_k). \quad (20)$$

Изложенный выше подход к задаче сглаживания (аппроксимации) сегмента кривой имеет преимущество по сравнению с традиционными методами сглаживания кривых, в которых набор измерений или выборка известны заранее. Этот подход дает возможность оценивать параметры кубической модели непосредственно в процессе поступления данных, что весьма актуально для приложений в системах реального времени. Рассмотрим применение изложенного метода для решения задачи (1) - (2).

3.3. Алгоритм LOCUS

Алгоритм LOCUS ("local cubic spline") предназначен для определения полного набора параметров кубического сплайна (включая узлы), оптимально приближающего кривую, заданную последовательностью точек на локальном отрезке. Оценки параметров находятся в два этапа: на первом этапе используются грубые оценки фиксированных параметров, DPT и РНК для

вычисления оценки свободного параметра $\hat{\theta}$ и числа точек n_θ , а на втором этапе эти величины используются для коррекции оценок фиксированных параметров $\hat{\theta}_\lambda$, $\hat{\theta}_l$ и $\hat{\theta}_{x_0}$. Алгоритм, по существу, является локально-оптимальным прогнозирующим фильтром третьего порядка. Его легко получить на основании формул (3), (13), (14), (15), (15а), (19) и (20) в следующем виде:

$$\begin{aligned} \hat{\theta}_0 &= -\tilde{\phi}_n^a / \mu m h^3; \\ \tilde{\phi}_n^a &= (\tilde{\mathbf{V}}_n, \mathbf{P}_n); \gamma_n = 6 / (\mu m (n+1)(2n+1)); \\ \hat{\theta}_n &= \hat{\theta}_{n-1} + \gamma_n [\tilde{\phi}_n^a h^{-3} - \hat{\theta}_{n-1} \mu m n]; \\ r_n &= \tilde{\phi}_n - (\hat{\mathbf{V}}_n, \mathbf{W}_n), \quad n = 1, 2, \dots; \quad n_\theta = n; \\ \hat{\mathbf{C}}^T &= (\mathbf{A}^T \mathbf{A})^{-1} \tilde{\mathbf{B}}^T; \quad \hat{\theta}_{x_0}. \end{aligned} \quad (21)$$

На рис.5 приведена структура алгоритма LOCUS. На схеме приведены величины, которые вычислялись в заданном окне заранее и заносились в LUT-таблицу.

Из (7) следует, что для восстановления $f(x)$ на отрезке $[x_l, x_R]$ с известными функциями $\mathbf{D}(\tau; \lambda, L)$, $\mathbf{Q}(\tau; \lambda, L)$ и заданной точностью T_f в TPS-модели используется девять параметров $x_0, \lambda, L, \hat{\theta}, \hat{\theta}_\lambda, \hat{\theta}_l, \hat{\theta}_{x_0}, x_l, x_R$. С целью экономии памяти их число можно сократить, используя формулы из [5] для вычисления оценок коэффициентов при x^2, x^1, x^0 в стандартной записи квадратичной

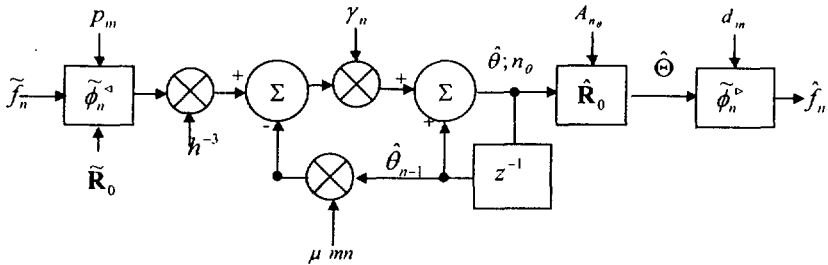


Рис.5. Структура алгоритма LOCUS

параболы через координаты реперных точек в виде

$$a_2 = (\lambda \hat{\theta}_l - L \hat{\theta}_\lambda) / (\lambda L H), \quad a = (\lambda^2 \hat{\theta}_l - L^2 \hat{\theta}_\lambda) / (\lambda L H), \quad a_0 = \hat{\theta}_{x_0}.$$

В результате на выходе алгоритма получим набор из шести параметров

$$\hat{\Theta} = \{\hat{\theta}, \hat{a}_2, \hat{a}_1, \hat{a}_0, x_l, x_R\},$$

полностью определяющий j -й локальный кубический сплайн в стандартной форме

$$S_j(x; \hat{\Theta}) = \hat{\theta} x^3 + \hat{a}_2 x^2 + \hat{a}_1 x + \hat{a}_0, \quad x \in [x_l, x_R], \quad j = 1, 2, \dots, M.$$

4. Примеры

Для исследования рабочих характеристик алгоритма (21) была создана программа (Maple V), в которой алгоритм LOCUS использовался для вычисления параметров локальных кубических сплайнов в задаче (1)-(2). Входные данные $\{\tilde{f}_k\}$ вычислялись на отрезке $[x_B, x_E]$ по заданной формуле $f = f(x)$ на сетке $x_k = x_B + kh, k = 1, 2, \dots, K_{\max}$ с добавлением случайных ошибок $e_k = rand()$, дисперсия которых σ_e^2 соответствовала разбросу ошибки в диапазоне $-\eta f_k \leq e_k \leq \eta f_k$, где $0 \leq \eta \leq \eta_{\max}$, так что $\tilde{f}_k = f_k(1 + e_k)$ (рис. 6, 7, 8). На выходе программы были получены наборы параметров $\{\hat{\phi}_i\}$, $j = 1, 2, \dots, M$, $M \geq 1$ кубических сплайнов S_j , локально сглаживающих входные точки $\{\tilde{f}_k\}$ с точностью, заданной порогом T_j .

Для оптимизации числа операций функции $p_m, d_m, i = 1, 2, 3; d_{1n}^2, d_{2n}^2, d_{1n}, d_{2n}, Q_n, \gamma_n$ и μn , $n = 1, 2, \dots, n_{\max}$ табулировались для заданных λ и L . Число n_{\max} определяет максимальную протяженность окна, в котором осуществлялся поиск локального сплайна S . Левая граница первого сплайна выбиралась в точке x_B , а правая граница находилась автоматически с помощью алгоритма LOCUS. При переходе к новому участку кривой правая граница предыдущего сплайна принималась за начало следующего и т.д.

Кроме параметров μ , m , и порога T_f на входе программы можно было задавать также порог T_θ с целью контроля величины коррекции свободного параметра. Порог "шумовой зоны" T_v определялся неявно, через параметры λ, L и начальную точку x_0 . Входные данные $\{\tilde{f}_k\}$ трансформировались в $\{\tilde{\phi}_k^a\}$ по формуле (3). Основные результаты работы программы представлены на рис. 6, 7 и 8. На графиках представлены гистограммы и разброс остатков (r_k, r_k^a) для исходных и преобразованных точек на всем промежутке $[x_B, x_E]$. На фоне входных данных сплошными линиями показаны графики локальных сплайнов S и узлов, найденных программой. Приведены также графики отклонений полученных TPS от истинных значений кривой $(f_k - S_{jk})$. В точках стыковки локальных сплайнов S_j наблюдается характерное поведение остатков r_k и r_k^a , которое вызвано частично неадекватностью модели форме кривой на данном участке при данном шаге квантования, а также влиянием "шумовых зон" на результат преобразования. Массив входных данных $\{\tilde{f}_k\}$, $x_k \in [x_B, x_E]$ генерировался с использованием показательных и тригонометрических функций.

Оценки среднеквадратичных погрешностей $\hat{\sigma}_r$ и $\hat{\sigma}_{r^a}$ для остатков $r_k = \tilde{f}_k - S_{jk}$ и $r_k^a = \tilde{\phi}_k^a - S_{jk}^a$ характеризуют точность сглаживания или аппроксимации кривых на исследуемом отрезке. Отношение объема данных на входе к объему данных на выходе N_{inp} / N_{out} определяет коэффициент сжатия информации в результате сглаживания. Приведены также суммы остатков $\sum_{k=1}^{K_{max}} r_k$ для всего отрезка $[x_B, x_E]$.

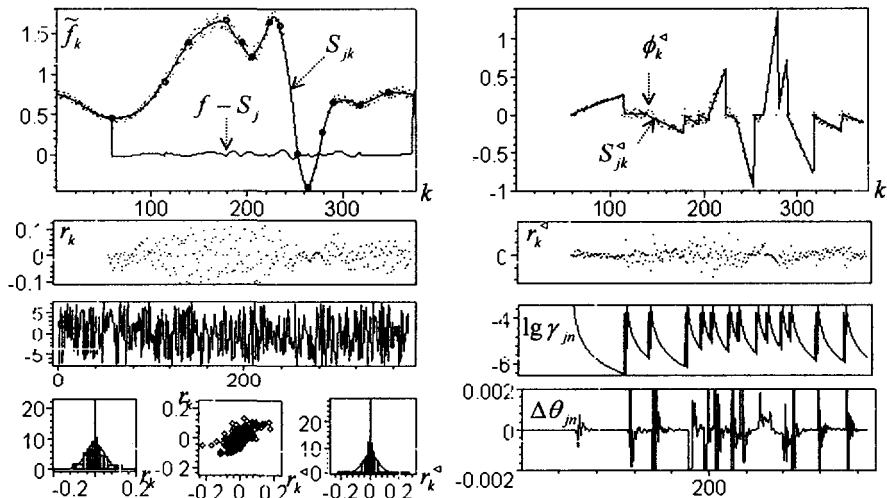


Рис.6. Графическое представление результатов алгоритма LOCUS ($\sigma_e^2 \neq 0$), (параметры на входе: $\sigma_e = 0.044$, $\mu = -49$, $m = -57$, $h = 0.125$ и $T_f = 0.12$; результат на выходе: $\hat{\sigma}_r = 0.0128$, $N_{inp} / N_{out} \approx 5.21$, $N_{spl} = 14$, $\sum r_k = 0.101$)

На примере графиков коррекции параметра $\hat{\theta}$ ($\Delta \hat{\theta}_n$) для S_j и коэффициентов усиления $\gamma_{jn} = \gamma_j(\mu, m; n)$ в логарифмическом масштабе можно наблюдать ход и скорость процесса адаптации TPS к кривой на каждом локальном отрезке.

Следует отметить, что зоны слежения (длины отрезков $[x_L, x_R]_j$) зависят от локальных свойств $f(x)$, шага квантования h , порога T_f и значений параметров μ, m . Например, при известных фиксированных параметрах и отсутствии шумов ($\sigma_e^2 = 0$), число локальных сплайнов уменьшается при увеличении порога T_f . На рис. 6, 7 и 8 показаны расположения узлов, найденных программой при аппроксимации и сглаживании.

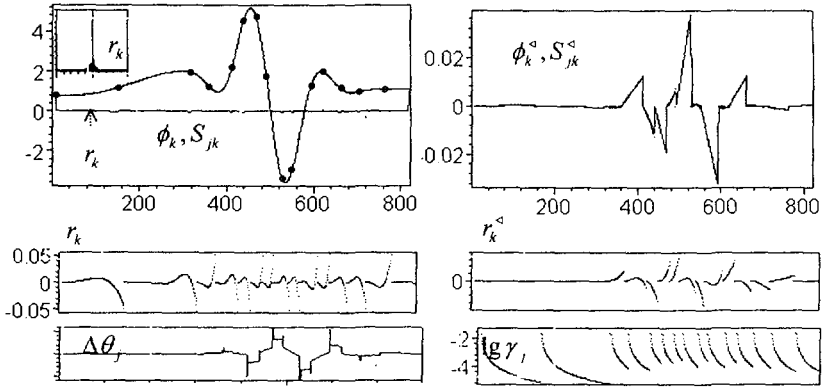


Рис.7. Приближение кривой локальными TPS ($\sigma_c^2 = 0$)

(параметры на входе: $\hat{\sigma}_c = 0$, $\mu = -3$, $m = -7$, $h = 0.05$ и $T_f = 0.05$;

результат на выходе: $\hat{\sigma}_r = 0.0128$, $N_{np} / N_{out} \approx 10.1$, $N_{spl} = 14$, $\sum r_k = -0.267$).

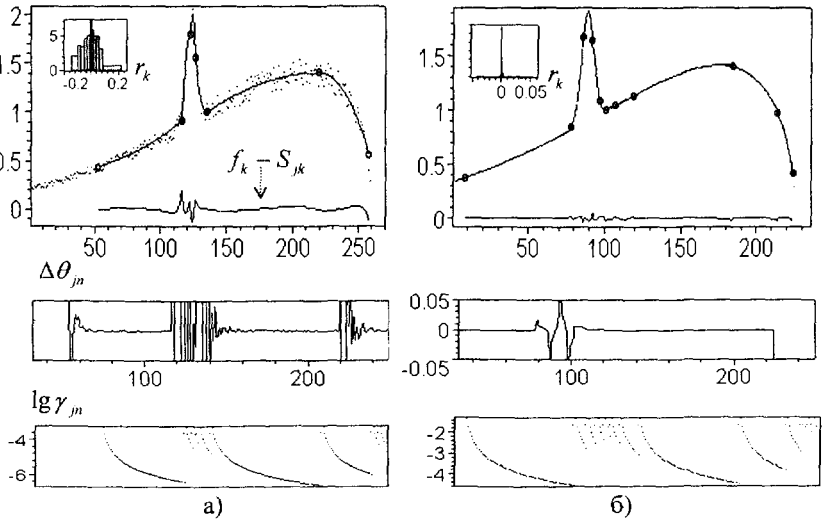


Рис.8. Сглаживание а) и аппроксимация б) $\tilde{f}(x)$ и $f(x)$ с использованием алгоритма LOCUS.

а) параметры на входе: $T_f = 0.2$, $m = -51$, $\mu = -37$, $h = 0.175$, $\hat{\sigma}_c = 0.059$;

результат на выходе: $\hat{\sigma}_r = 0.067$, $N_{np} / N_{out} \approx 6.5$, $N_{spl} = 6$, $\sum r_k = -0.45$.

б) параметры на входе: $T_f = 0.2$, $m = -7$, $\mu = -3$, $h = 0.175$, $\hat{\sigma}_c = 0$;

результат на выходе: $\hat{\sigma}_r = 0.007$, $N_{np} / N_{out} \approx 3.3$, $N_{spl} = 10$, $\sum r_k = -0.03$

5. Заключение

В работе предложен новый адаптивный метод оптимально-локального приближения и сглаживания функций 3-точечными кубическими сплайнами и построен простой итерационный алгоритм локального сглаживания третьего порядка (LOCUS). Скорость сходимости итераций характеризуется коэффициентом усиления $\gamma_n \propto 1/n^3$. В алгоритме использована схема последовательного определения параметров модели, что позволило значительно сократить число операций и существенно уменьшить объем рабочей памяти, необходимой для хранения исходных и промежуточных данных. На основе предложенного в работе математического аппарата и полученных результатов обработки смоделированных данных можно выделить следующие особенности алгоритма (21), характеризующие его эффективность, работоспособность и технологичность:

- a) точность приближения третьего порядка;
- b) возможность обработки данных в режиме их поступления;
- c) простая вычислительная схема;
- d) гарантированная сходимость итераций;
- e) устойчивость к случайным ошибкам;
- f) параметрическая настройка;
- g) автоматическое определение граничных узлов;
- h) весовые функции известны;
- i) возможность сжатия данных;
- j) работа с функциями, заданными таблично и аналитически;
- k) табулирование весовых функций в пределах выбранного окна.

Эти свойства обеспечивают такие характеристики алгоритма, как *адаптивность* (b, d, e, f, g), *устойчивость* (c, e), *точность* (a, e, h), *скорость* (c, d, k), *гибкость* (a, b, f, h, j, k), *эффективность* (b, c, d, h, i, k) и *простоту реализации* (c, f, k). Например, оценка скорости работы алгоритма на этапе адаптации, выраженная в числе арифметических операций на один итерационный цикл, составляет примерно 18 коротких операций, что в два раза меньше числа операций, необходимых для реализации рекурсивного алгоритма наименьших квадратов третьего порядка [1]. Эффективность алгоритма можно оценить как его скоростью, так и ресурсом памяти, необходимой для хранения данных, программ и рабочих полей, которые в нашем случае минимальны, так как вычисление оценок параметров выполняется в режиме поступления данных и не требует дополнительной памяти под рабочие поля и для хранения выборки в полном объеме. Точность алгоритма обеспечивается порядком выбранной модели, точностью весовых коэффициентов, а также оптимальностью критериев, используемых для вычисления оценок.

Возможности предложенного метода показаны на примере создания алгоритма для кусочно-кубической аппроксимации функций в режиме адаптации. Рабочие характеристики сглаживающего алгоритма исследованы программным путем. Устойчивость метода к ошибкам, его надежность и эффективность подтверждены результатами обработки сгенерированных

данных. Алгоритм использует небольшие ресурсы памяти и ориентирован на применение в таких областях, как цифровая обработка сигналов, обработка контурных изображений, распознавание треков, а также при численном решении многих практических задач.

Литература

- [1] C.F.N. Cowan and P.M. Grant, eds., *Adaptive Filters*, Prentice-Hall, Englewood Cliffs, 1985.
(К.Ф.Н. Коуэн, П.М. Грант (ред.), *Адаптивные фильтры*, Мир, М., 1988).
R.W. Hamming, *Digital Filters*, Prentice-Hall, Englewood Cliffs, 1977.
(Р.В.Хемминг, *Цифровые фильтры*, "Советское радио", М., 1980).
- [2] N.D. Dikoussar, *Comput. Phys. Commun.*, 99 (1997) 235-254.
- [3] M.T. Wasan, *Stochastic Approximation*, Cambridge University Press, N.Y., 1969.
(М.Т. Вазан, *Стохастическая аппроксимация*, Мир, М., 1972).
- [4] G.A.F. Seber, *Linear Regression Analysis*, John Wiley and Sons, N.Y., 1977.
(Дж. Себер, *Линейный регрессионный анализ*, Мир, М., 1980.)
- [5] N.D. Dikoussar, *Comput. Phys. Commun.*, 79 (1994), 39-51.
- [6] Т ö р ö k Cs., *Determining the degree of polynomial in regression model*, Slovak Statistical Days 1995, Bratislava, 52-56.
Т ö р ö k Cs., E5-97-254, Dubna, 1997.
- [7] M.Robbins and S.Monroe, *A Stochastic Approximation Method*, *Annals of Mathematical Statistics*, 22 (1951) 400-407.
- [8] M.V. Avdeev and N.D. Dikoussar, *Proc. of 9th Intern. Conf. Computational Modeling and Computing in Physics*, JINR, D5, 11-97-112, Dubna, 1997, p.82-90.
- [9] В.Н.Фомин и др., *Адаптивное управление динамическими объектами*, Наука, М., 1981.
- [10] А.А. Красовский (ред.), *Справочник по теории автоматического управления*, Наука, М., 1987.
- [11] M. Regler (ed.), *Data analysis techniques for high-energy physics experiments*, Cambridge University Press, 1990. (*Методы анализа данных в физическом эксперименте*, Мир, М., 1993).
- [12] Г.Н. Зорин и др. *Препринт ОИЯИ*, 18-95-529, Дубна, 1995.
С.П. Третьякова, ЭЧАЯ, 1992, т. 23 вып. 2, с. 364.

Рукопись поступила в издательский отдел
15 июня 1999 года.