

**РОБАСТНОЕ ОБУЧЕНИЕ РАДИАЛЬНО-БАЗИСНЫХ СЕТЕЙ**

**Ключевые слова:** *нейронная сеть, робастное обучение, нелинейный объект, идентификация, базисная функция.*

**ВВЕДЕНИЕ**

Радиально-базисные сети (РБС) находят широкое применение в задачах идентификации, управления, распознавания образов и др. [1–3]. Наличие достаточно эффективных рекуррентных алгоритмов обучения данных сетей позволяет применять их для решения указанных задач в реальном времени [4–8]. На практике как обучение сети, так и ее дальнейшее использование приходится осуществлять с учетом присутствующих помех измерений.

Большинство существующих в настоящее время методов обучения основано на использовании жестких и труднопроверяемых условий, связанных с гипотезой нормальности закона распределения помех и обоснованных ссылками на центральную предельную теорему. Как известно [9], нормальным законом плотности распределения описываются помехи, присутствующие в измерениях, проводимых при абсолютной стабильности условий измерения, законом Лапласа, имеющем более длинные «хвосты» — помехи, возникающие при максимальной нестабильности условий. Соответственно алгоритмы обучения в случае гауссовских помех основаны на методе наименьших квадратов (МНК), а в случае помех, распределенных по закону Лапласа, — на методе наименьших модулей (МНМ). Оба метода являются оптимальными в своих условиях, и решения, получаемые с их помощью, могут существенно отличаться. Кроме того, так как на практике эти крайние случаи реализуются чрезвычайно редко, ни закон Гаусса, ни закон Лапласа, как правило, не выполняются.

В [10, 11] рассмотрены некоторые типы классов распределений, встречающиеся при решении практических задач:  $P_1$  — класс невырожденных распределений,  $P_2$  — класс распределений с ограниченной дисперсией,  $P_5$  — класс финитных распределений (помеха ограничена по абсолютной величине, а какие-либо сведения о плотности ее распределения отсутствуют),  $P_3$ ,  $P_4$  и  $P_6$  — классы приближенно нормальных, приближенно равномерных и приближенно финитных распределений соответственно, описываемых моделью Тьюки–Хьюбера [12–14]

$$p(x) = (1 - \varepsilon)p_0(x) + \varepsilon q(x), \quad (1)$$

где  $p_0(x)$  — плотность соответствующего основного распределения;  $q(x)$  — плотность засоряющего (произвольного) распределения;  $\varepsilon \in [0, 1]$  — параметр, характеризующий степень засорения основного распределения.

Для всех этих классов найдены наименее благоприятные, т.е. минимизирующие фишеровскую информацию, распределения. Так, минимум фишеровской информации для класса  $P_1$  дает распределение Лапласа  $p^*(\xi) = L(0, s_\xi)$ , для класса  $P_2$  — распределение Гаусса  $p^*(\xi) = N(0, \sigma^2)$ , для классов финитных

распределений наименее благоприятной плотностью является

$$p^*(\xi) = \begin{cases} \frac{1}{l} \cos^2 \frac{\pi\xi}{2l} & \text{при } |\xi| \leq l, \\ 0 & \text{при } |\xi| > l. \end{cases}$$

Для  $\varepsilon$ -засоренных вероятностных распределений (1) плотность распределения  $p^*$ , дающего минимум фишеровской информации, содержит некоторую центральную область  $p = (1 - \varepsilon)p_0(\xi)$  и «хвосты» с экспоненциально убывающей плотностью  $p_0(\xi) = ce^{-\lambda|\xi|}$ .

Использование этих распределений позволило получить нелинейные оценки огрубленного или робастного метода максимального правдоподобия, работоспособные практически для любых распределений помех.

Рекуррентные одномерные алгоритмы оценивания, базирующиеся на методе стохастической аппроксимации и требующие минимальных вычислительных затрат, обеспечивая при этом ту же асимптотическую точность оценивания, что и нереккуррентные, приведены в [10]. В работе [11] разработаны многомерные абсолютно оптимальные алгоритмы оценивания параметров линейной регрессионной модели, имеющие структуру, аналогичную структуре рекуррентного МНК, и работоспособные после количества тактов, превышающего число неизвестных параметров.

Цель настоящей статьи — исследование робастных проекционных алгоритмов, с помощью которых обучение можно осуществлять, начиная с первого такта поступления информации.

#### ПОСТАНОВКА ЗАДАЧИ

Рассмотрим нелинейный объект, описываемый уравнением

$$y(k) = f^*(\mathbf{w}^*, \mathbf{x}(k)) + \xi(k), \quad (2)$$

где  $f^*(\cdot)$  — неизвестная нелинейная функция;  $\mathbf{x}(k) = (x_1(k), x_2(k), \dots, x_N(k))^T$  — вектор независимых переменных;  $\mathbf{w}^* = (w_1^*, w_2^*, \dots, w_N^*)^T$  — вектор неизвестных параметров;  $\xi(k)$  — помеха.

Представление нелинейности  $f^*(\cdot)$  радиально-базисной сетью

$$f(\mathbf{x}(k)) = \mathbf{c}^T \Phi(k), \quad (3)$$

где  $\Phi(k) = (1, \varphi_1(k), \varphi_2(k), \dots, \varphi_N(k))$  — вектор выбранных базисных функций (БФ);  $\mathbf{c} = (c_0, c_1, \dots, c_N)^T$  — вектор весовых коэффициентов, позволяет свести задачу идентификации к задаче обучения сети, которое должно обеспечить определение ее параметров, в частности вектора  $\mathbf{c}$ .

При использовании гауссовских функций

$$\Phi_i(\mathbf{x}) = \exp \left\{ -\frac{\|\mathbf{x} - \mu_i\|^2}{\sigma_i^2} \right\}, \quad (4)$$

в качестве базисных, где  $\mu_i, \sigma_i$  — центры и радиусы БФ соответственно;  $\|\cdot\|$  — евклидова норма, в результате обучения помимо оценок коэффициентов  $c_i$  должны быть получены и оценки  $\mu_i$  и  $\sigma_i$ .

Введя вектор оценок настраиваемых параметров

$$\mathbf{w} = (c_0, c_1, \mu_1^T, \sigma_1, \dots, c_N, \mu_N^T, \sigma_N)^T, \quad (5)$$

нейросетевую модель (3) можно представить в виде

$$\hat{y}(k) = f(\mathbf{w}, \mathbf{x}(k)). \quad (6)$$

Задача обучения сети заключается в минимизации некоторой выпуклой функции потерь  $F[e]$ , где  $e(k) = y(k) - \hat{y}(k)$ .

Робастные оценки определяются из условия

$$\mathbf{w}(k) = \arg \min I_k(\mathbf{w}), \quad I_k(\mathbf{w}) = \sum_{i=1}^k F[e(\mathbf{w}, \mathbf{x}(k))], \quad F(e) = -\ln p(\xi)|_{\xi=e} \quad (7)$$

и являются оптимальными в минимаксном смысле на соответствующих классах распределений.

Оптимальная функция потерь, т.е. функция, минимизирующая асимптотическую матрицу ковариации ошибок оценивания, равна логарифму плотности распределения помехи при  $\xi = e(\mathbf{x}, \mathbf{w})$ , взятой с обратным знаком,

$$F_0[e(\mathbf{x}, \mathbf{w})] = -\ln p_0(\xi)|_{\xi=e(\mathbf{x}, \mathbf{w})},$$

т.е. логарифмической функции неправдоподобия.

Таким образом, оптимальной функцией потерь для класса  $P_1$  будет модульная, для класса  $P_2$  — квадратичная, для класса  $P_5$  в виде

$$F_0[e(\mathbf{x}, \mathbf{w})] = -\ln \cos\left(\frac{\pi e}{c}\right).$$

Им соответствуют кривые 3, 1 и 2 на рис. 1.

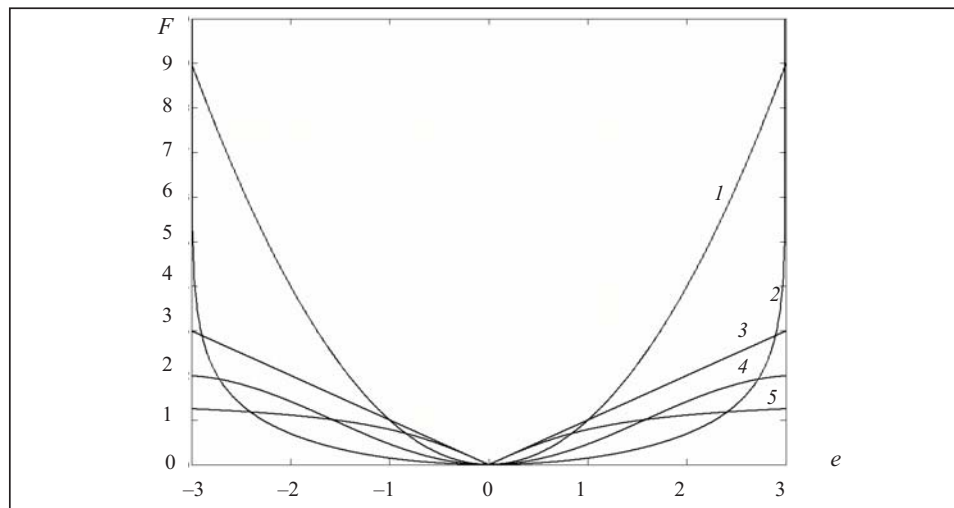


Рис. 1

Для  $\varepsilon$ -засоренных вероятностных распределений функция  $F_0$  является нелинейной на некотором интервале, определяемом параметрами помех, и линейной вне этого интервала.

В силу того, что модульный критерий позволяет получить оценку, менее чувствительную к «хвостам» распределения помехи, чем МНК-оценка, представляют интерес такие разновидности модульного критерия, как функционал

А. Форсайта, представленный на рис. 2,

$$F[e(k)] = |e(k)|^\lambda, \quad (8)$$

и функционал вида

$$F[e(k)] = \arctg |e(k)|^\lambda, \quad (9)$$

где  $0 < \lambda < 2$  (рис. 3).

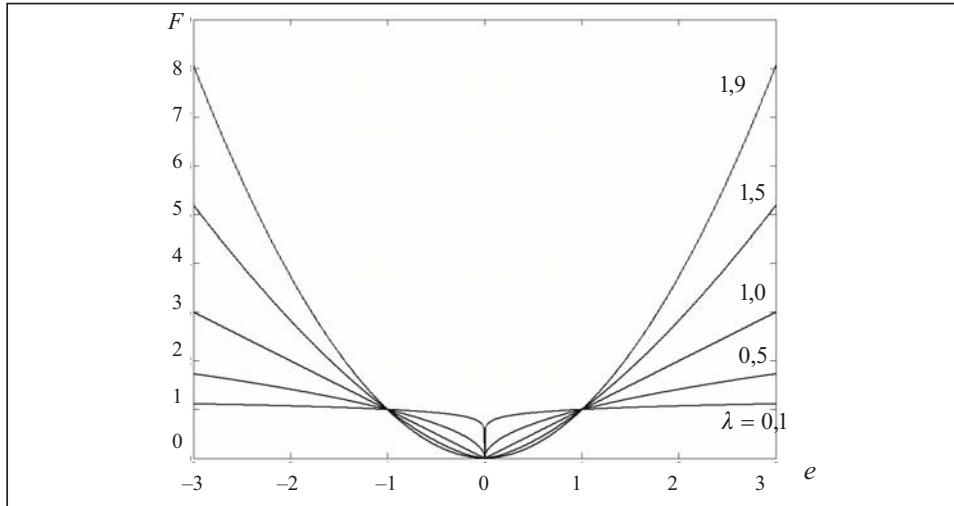


Рис. 2

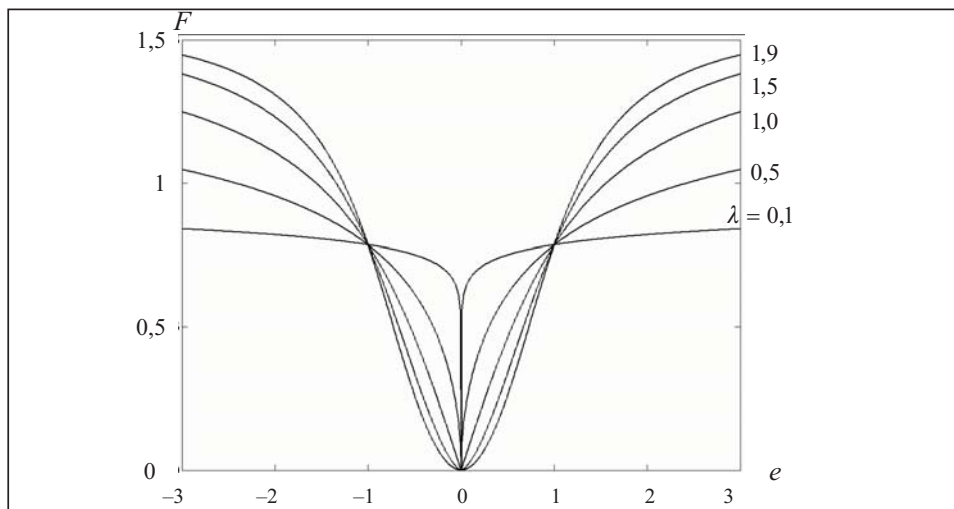


Рис. 3

Кривые, соответствующие функционалам (8) и (9) при  $\lambda = 1$ , обозначены на рис. 1 цифрами 3 и 5; кривая 4, соответствующая функционалу Д. Эндрюса

$$F[e(k)] = \begin{cases} 1 - \cos\left(\frac{e}{c}\right), & |e| \leq \pi c, \\ 0, & |e| > \pi c, \end{cases}$$

занимает некоторое промежуточное положение между кривыми 3 и 5.

## ОПТИМАЛЬНЫЕ АЛГОРИТМЫ ОБУЧЕНИЯ

Если плотность распределения помех  $p_0(\xi)$  полностью неизвестна, а известно лишь, что она принадлежит некоторому классу  $p_k$ , то для наименее благоприятной на этом классе плотности распределения  $p^*(\xi)$  можно получить соответствующий оптимальный алгоритм. Если  $p_0(\xi) = p^*(\xi)$ , этот алгоритм будет абсолютно оптимальным для класса  $p_k$ .

Оптимальные алгоритмы обучения сети представляют собой градиентные процедуры вида

$$\mathbf{w}(k) = \mathbf{w}(k-1) - H(k)\nabla F_0[e(k)], \quad (10)$$

где  $H(k)$  — матрица усиления.

Учитывая (4) и то, что  $\nabla F_0[e(k)] = F'[e(k)]\nabla f(k)$ , где

$$\nabla f(k) = [1, \Phi_1(\mathbf{x}(k)), 2\Phi_1(\mathbf{x}(k))c_1\sigma_1^{-2}(\mathbf{x}(k) - \mu_1)^T, 2\Phi_1(\mathbf{x}(k))c_1\sigma_1^{-3}\|\mathbf{x}(k) - \mu_1\|^2, \dots, \Phi_N(\mathbf{x}(k)), 2\Phi_N(\mathbf{x}(k))c_N\sigma_N^{-2}(\mathbf{x}(k) - \mu_N)^T, 2\Phi_N(\mathbf{x}(k))c_N\sigma_N^{-3}\|\mathbf{x}(k) - \mu_N\|^2]^T,$$

можно записать оптимальные алгоритмы обучения следующим образом:

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \frac{P(k-1)\nabla f(k)}{\gamma + \nabla f^T(k)P(k-1)\nabla f(k)} F'[e(k)], \quad (11)$$

$$P(k) = P(k-1) - \frac{P(k-1)\nabla f(k)\nabla f^T(k)P(k-1)}{\gamma + \nabla f^T(k)P(k-1)\nabla f(k)}. \quad (12)$$

Здесь

$$F'[e(k)] = \begin{cases} e(k) & \text{для } p_0(\xi) = N(0, \sigma_\xi^2), \\ \text{sign}(e(k)) & \text{для } p_0(\xi) = L(0, s_\xi), \\ \frac{2e(k)}{s_\xi^2 + e^2(k)} & \text{для } p_0(\xi) = C(0, s_\xi); \end{cases} \quad (13)$$

$$\text{sign}(e(k)) \quad \text{для } p_0(\xi) = L(0, s_\xi), \quad (14)$$

$$\frac{2e(k)}{s_\xi^2 + e^2(k)} \quad \text{для } p_0(\xi) = C(0, s_\xi); \quad (15)$$

$$\gamma = \begin{cases} 1 & \text{для } p_0(\xi) = N(0, \sigma_\xi^2), \\ s_\xi & \text{для } p_0(\xi) = L(0, s_\xi), \\ s_\xi^2 & \text{для } p_0(\xi) = C(0, s_\xi). \end{cases} \quad (16)$$

$$s_\xi \quad \text{для } p_0(\xi) = L(0, s_\xi), \quad (17)$$

$$s_\xi^2 \quad \text{для } p_0(\xi) = C(0, s_\xi). \quad (18)$$

Как следует из приведенных соотношений, если алгоритм МНК (11)–(13), (16) инвариантен к параметру масштаба нормального распределения  $\sigma_\xi^2$ , то оценки МНМ (11), (12), (14), (17) и оценка, соответствующая распределению Коши, зависят от параметра  $s_\xi$ . При неизвестной величине  $s_\xi$  в этих алгоритмах следует использовать оценку  $\hat{s}$ , получаемую, например, с помощью алгоритма стохастической аппроксимации.

Для  $\varepsilon$ -засоренных вероятностных распределений оптимальные алгоритмы обучения представляют собой комбинированные процедуры вида

$$\mathbf{w}(k) - \mathbf{w}(k-1) = \begin{cases} -H_1(k)F'[e(k)]\nabla f(k) & \text{при } |e(k)| \leq \Delta, \\ -H_2(k)\nabla f(k)\text{sign } e(k) & \text{при } |e(k)| > \Delta, \end{cases} \quad (19)$$

где  $H_1(k)$ ,  $H_2(k)$  — матрицы усиления.

## ПРОЕКЦИОННЫЕ АЛГОРИТМЫ ОБУЧЕНИЯ

Рассмотренные выше алгоритмы обучения будут работоспособными после  $k \geq N + 1$  тактов. Если  $k < N + 1$ , обучение сети может осуществляться с помощью проекционных алгоритмов [15]. Следует также отметить, что применение этих алгоритмов целесообразно и в случае, когда исследуемый объект является нестационарным и его параметры изменяются с течением времени.

$l$ -шаговый проекционный алгоритм обучения имеет вид

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \gamma(k) \nabla f_l(k) [\nabla f_l^T(k) \nabla f_l(k)]^{-1} F'_l[e(k)], \quad (20)$$

где  $F'_l[e(k)] = (F'[e(k)], 0, 0, \dots, 0)^T$ ;  $\nabla f_l(k) = (\nabla f(k), \nabla f(k-1), \dots, \nabla f(k-l+1))$  — матрицы  $(N+1) \times l$ ;  $\gamma(k)$  — параметр, удовлетворяющий условиям Дворецкого.

Более удобная в вычислительном отношении процедура, использующая рекуррентное вычисление матрицы  $[\nabla f_l^T(k) \nabla f_l(k)]^{-1}$  вместо ее непосредственного обращения, может быть получена с учетом правил формирования матриц  $\nabla f_l(k-1) = (\nabla f_{l-1}(k-1) \vdots \nabla f(k-l))$ ,  $\nabla f_l(k) = (\nabla f(k) \vdots \nabla f_{l-1}(k-1))$ .

В этом случае можно записать

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \gamma(k) \frac{R_{l-1}(k-1)}{\nabla f^T(k) R_{l-1}(k-1) \nabla f(k)} F'_l[e(k)], \quad (21)$$

где

$$\begin{aligned} R_{l-1}(k-1) &= I - \nabla f_{l-1}(k-1) [f_{l-1}^T(k-1) \nabla f_{l-1}(k-1)]^{-1} \nabla f_{l-1}^T(k-1) = \\ &= I - f_{l-1}^+(k-1) f_{l-1}^T(k-1). \end{aligned}$$

Здесь  $\nabla f_{l-1}^+(k-1)$  — матрица, псевдообратная к  $\nabla f_{l-1}(k-1)$ .

Матрица  $R_l(k)$  также вычисляется рекуррентно. Если получение соотношений для ее вычисления при поступлении новой информации достаточно тривиально, то для реализации процедуры сброса следует воспользоваться теоремой Гревилля и следующими соотношениями между матрицами  $\nabla f_{l+1}^+(k)$  и  $\nabla f_l^+(k)$  [16]:

$$(\nabla f_l^+(k) \vdots 0) = \nabla f_{l+1}^+(k) (I - \mathbf{g} \mathbf{K}^T(k)).$$

Здесь для рассматриваемого случая  $l < N$

$$\mathbf{K}(k) = \frac{(I - \nabla f_{l+1}(k) \nabla f_{l+1}^+(k)) \mathbf{g}}{\mathbf{g}^T (I - \nabla f_{l+1}(k) \nabla f_{l+1}^+(k)) \mathbf{g}},$$

$\mathbf{g} = (0, 0, \dots, 1)^T$  — вектор  $l \times 1$ .

Проводя несложные вычисления, получим, что при поступлении новой информации на  $k$ -м такте матрица  $R_l(k)$  пересчитывается по формуле

$$R_l(k) = R_{l-1}(k-1) + \frac{R_{l-1}(k-1) \nabla f(k) \nabla f^T(k) R_{l-1}(k-1)}{\nabla f^T(k) R_{l-1}(k-1) \nabla f(k)}, \quad (22)$$

а при сбросе устаревшей информации о  $(k-l+1)$ -м такте — по формуле

$$R_l(k+1) = R_{l+1}(k+1) + \frac{\nabla \hat{f}(k-l+1) \nabla \hat{f}^T(k)}{\|\nabla \hat{f}(k-l+1)\|^2}, \quad (23)$$

где

$$\hat{\nabla} f(k-l+1) = U_{l+1}(k+1) \nabla f(k-l+1), \quad R_0 = I,$$

$$U_{l+1}(k+1) = (\nabla f_{l+1}^T(k+1) \nabla f_{l+1}(k+1))^+.$$

Входящая в (22), (23) матрица  $U_l(k)$ , в свою очередь, при поступлении новой информации и сбросе устаревшей вычисляется соответственно по следующим формулам:

$$\begin{aligned} U_{l+1}(k+1) = & U_l(k) - \frac{(U_l(k) + R_l(k)) \nabla f(k) \nabla f^T(k) U_l(k)}{\nabla f^T(k+1) U_l(k) \nabla f(k+1)} + \\ & + \frac{R_l(k) \nabla f(k+1) \nabla f^T(k+1) R_l(k)}{\nabla f^T(k+1) U_l(k) \nabla f(k+1)} \times \\ & \times \frac{R_l(k) \nabla f(k+1) \nabla f^T(k+1) U_l(k) \nabla f(k) \nabla f^T(k+1) R_l(k)}{\|R_l(k) \nabla f(k+1)\|^2}; \end{aligned} \quad (24)$$

$$\begin{aligned} U_l(k+1) = & U_{l+1}(k+1) - \\ & - \frac{U_{l+1}(k+1) \hat{\nabla} f(k-l+1) \hat{\nabla} f(k-l+1) + \hat{\nabla} f(k-l+1) \hat{\nabla} f^T(k) U_{l+1}(k+1)}{\|\hat{\nabla} f(k-l+1)\|^2} + \\ & + \frac{\hat{\nabla} f(k-l+1) U_{l+1}(k+1) \hat{\nabla} f(k-l+1)}{\|\hat{\nabla} f(k-l+1)\|^4} \hat{\nabla} f(k-l+1) \hat{\nabla} f^T(k-l+1). \end{aligned} \quad (25)$$

Таким образом, процедуры (21), (22), (24) соответствуют накоплению, а (21), (23), (25) — сбросу устаревшей информации.

#### МОДЕЛИРОВАНИЕ

Рассмотрена задача идентификации нелинейного динамического объекта при наличии различных помех в выходных сигналах. Объект описывался уравнением

$$y(k) = 0,725 \sin \left( \frac{16u(k-1) + 8y(k-1)}{3 + 4u^2(k-1) + 4y^2(k-1)} \right) + 0,2u(k-1) + 0,2y(k-1) + \xi, \quad (26)$$

где  $u(k)$  — входной сигнал, представляющий стационарную случайную последовательность с равномерным законом распределения в интервале  $[-1,1]$ , генерируемую датчиком случайных чисел;  $\xi$  — помеха измерений с плотностью вероятности (1).

При исследовании использовано 10 000 обучающих пар. Количество нейронов для сетей во всех экспериментах было взято равным 15. Общее количество неизвестных параметров  $N$ , определяемых в процессе обучения сетей, равно 61. Некоторые результаты сравнительного анализа рекуррентных алгоритмов МНК, МНМ, комбинированных алгоритмов, построенных на их основе, и двадцатишагового проекционного алгоритма ( $l=20$ ) при использовании различных функционалов приведены в табл. 1. Здесь представлены значения среднеквадра-

точной ошибки, вычисленной по формуле

$$\theta = \frac{\sqrt{\sum_{i=1}^{2500} (y^*(i) - \hat{y}(i))^2}}{2500}, \quad (27)$$

где  $y^*$  — эталонное значение выходного сигнала в случае отсутствия помех измерений;  $\hat{y}$  — реальный выходной сигнал сетей.

**Таблица 1**

Основное распределение	Лапласа ( $M = 0,$ $s_{\xi} = 8$ )	Нормальное ( $M = 0,$ $\sigma = 0,4$ )	Равномерное ( $M = 0,$ $\sigma = 0,8$ )	Лапласа ( $M = 0,$ $s_{\xi} = 8$ )		Нормальное ( $M = 0,$ $\sigma = 0,4$ )		Без помехи	
Засоряющее распределение	–	–	–	Нормальное ( $M = 0,$ $\sigma = 2$ )		Нормальное ( $M = 0,$ $\sigma = 2$ )		–	
$\varepsilon$	0	0	0	0,05	0,1	0,05	0,1	–	
МНК	2,1020	3,0170	3,6120	6,9518	8,0786	6,6651	8,3679	0,6403	
МНМ	1,8929	3,4673	3,1307	2,3542	3,0718	4,3433	6,2069	1,1007	
Проекционный алгоритм	$\ e(k)\ ^2$	1,9735	2,8915	2,1381	4,6996	5,5332	5,6818	5,7541	0,6867
	$ e(k) $	1,7208	2,1284	2,7486	1,7290	2,9641	3,4821	4,6333	1,0394
	$\arctg  e(k) $	1,8083	2,6765	2,7507	1,7075	2,1231	3,5512	4,1693	0,9796
	$-\ln \cos(e(k))$	1,8541	1,9581	1,8949	1,9299	5,1369	4,5750	5,3119	0,8727
	$1 - \cos\left(\frac{e(k)}{c}\right)$ $c = 0,5$	1,7109	2,6738	2,7633	1,9723	4,0251	2,9376	4,2039	0,7165
МНК+МНМ ( $e_{\text{доп}} = 1,2$ )	1,6343	2,9408	2,5359	2,9170	3,6414	5,2157	6,4183	–	
Две сети: ( $-4\varepsilon + 0,6$ ) * МНК + ( $4\varepsilon + 0,4$ ) * МНМ	–	–	–	3,7110	4,3165	5,9186	6,2804	–	

В последних двух строках таблицы приведены результаты моделирования работы алгоритмов — комбинации МНК и МНМ, причем в первом алгоритме использовался МНК, когда ошибка обучения не превышала некоторого заранее заданного значения  $e_{\text{доп}}$  (так как выходной сигнал объекта изменяется от  $-1$  до  $1$ , в эксперименте было принято  $e_{\text{доп}} = 1,2$ ), и МНМ — при нарушении этого условия. Во втором алгоритме использованы две сети, обученные с помощью различных алгоритмов, а для получения значения выходного сигнала сети осуществлялось взвешенное суммирование сигналов обеих сетей с коэффициентом, зависящим от степени засорения  $\varepsilon$ .

#### ЗАКЛЮЧЕНИЕ

В основе робастных методов обучения лежат те или иные критерии, позволяющие «подавлять» нежелательные свойства выборок данных, используемых в алгоритмах. Естественной платой за надежность получаемых оценок является некоторая потеря эффективности по сравнению с оптимальными методами обучения при известных статистических характеристиках обрабатываемых данных. Однако если информация о виде плотностей распределения отсутствует, то робастные методы имеют несомненное преимущество, а говорить об их оптимальности или неоптимальности нет никаких оснований [17].

Таким образом, при наличии информации о виде плотности распределения помех выбор алгоритма обучения можно осуществить достаточно просто. Если



такой информации нет, то в качестве функции потерь целесообразно применить какую-либо из рассмотренных разновидностей модульных критериев, а в качестве алгоритма обучения — многошаговые проекционные процедуры. При этом, однако, остается открытым вопрос о выборе оптимального объема  $l$  используемой в этих алгоритмах информации. Кроме того, представляет интерес дальнейшее исследование комбинированных алгоритмов обучения, которые получают в результате минимизации различных функционалов.

#### СПИСОК ЛИТЕРАТУРЫ

1. Moody J., Darken C. Fast learning in networks of locally-tuned processing units // *Neural Computation*. — 1989. — 1. — P. 281–294.
2. Хайкин С. Нейронные сети: полный курс. — М.: Изд. дом «Вильямс», 2006. — 1104 с.
3. Бодянский Е.В., Руденко О.Г. Искусственные нейронные сети: архитектуры, обучение, применения. — Харьков: ТЕЛЕТЕХ, 2004. — 372 с.
4. Spooner J.T., Passino K.M. Decentralized adaptive control of nonlinear systems using radial basis neural networks // *IEEE Trans. Automat. Control*. — 1999. — 44, N 11. — P. 2050–2057.
5. Yu D.L., Yu D.W. A new structure adaptation algorithm for RBF networks and its application // *Neural Comput.&Application*. — 2007. — 16. — P. 91–100.
6. Shilling R.J., Carroll J.J., Al-Ajlouni A.F. Approximation of nonlinear systems with radial basis function neural networks // *IEEE Trans. Neural Networks*. — 2001. — 12, N 6. — P. 1–15.
7. Руденко О.Г., Бессонов А.А. Идентификация нелинейных нестационарных объектов в реальном времени с помощью радиально-базисных сетей // *Кибернетика и системный анализ*. — 2003. — № 6. — С. 177–185.
8. Руденко О.Г., Бессонов А.А. Адаптивное управление многомерными нелинейными объектами на основе радиально-базисных сетей // Там же. — 2005. — № 2. — С. 168–176.
9. Вапник В.М. Восстановление зависимостей по эмпирическим данным. — М.: Наука, 1979. — 448 с.
10. Цыпкин Я.З., Поляк Б.Т. Огрубленный метод максимального правдоподобия // *Динамика систем*. — Горький, 1977. — Вып. 12. — С. 22–46.
11. Цыпкин Я.З. Основы информационной теории идентификации. — М.: Наука, 1984. — 320 с.
12. Хьюбер П. Робастность в статистике. — М.: Мир, 1984. — 304 с.
13. Мудров В.И., Кушко В.Л. Методы обработки измерений (Квазиправдоподобные оценки). — М.: Сов. радио, 1976. — 192 с.
14. Смоляк С.А., Титаренко Б.П. Устойчивые методы оценивания: статистическая обработка неоднородных совокупностей. — М.: Статистика, 1980. — 208 с.
15. Либероль Б.Д., Руденко О.Г. О влиянии помех измерений на свойства проекционных алгоритмов идентификации // *Доп. НАН України*. — 1995. — № 3. — С. 28–30.
16. Альберт А. Регрессия, псевдоинверсия и рекуррентное оценивание. — М.: Наука, 1977. — 224 с.
17. Reу W.J.J. Robust statistical methods. — Berlin; Heidelberg; New York: Springer, 1978. — 128 p.

*Поступила 07.07.2009*