

УДК 004.8.023 : 681.51.015

ИДЕНТИФИКАЦИЯ НЕЧЁТКИХ СИСТЕМ НА ОСНОВЕ НЕПРЕРЫВНОГО АЛГОРИТМА МУРАВЬИНОЙ КОЛОНИИ*

И. А. Ходашинский, П. А. Дудин

*Государственное образовательное учреждение высшего профессионального образования
«Томский государственный университет систем управления и радиоэлектроники»,
634050, г. Томск, просп. Ленина, 40
E-mail: hodashn@rambler.ru*

Для идентификации нечётких систем впервые предложен непрерывный алгоритм муравьиной колонии. На его основе построено пять нечётких систем, аппроксимирующих нелинейные функции от одной, двух и трёх переменных. Исследовано влияние параметров алгоритма на ошибку аппроксимации. Сравнительный анализ с другими алгоритмами идентификации показал преимущество предложенного.

Ключевые слова: идентификация, нечёткие системы, непрерывный алгоритм муравьиной колонии.

Введение. Задачей идентификации системы является построение модели, описывающей отображение «вход—выход» системы. В нечёткой идентификации это отображение задаётся в виде нечёткой системы. Основное преимущество таких систем — возможность задания на их основе сложных и неопределённых отношений достаточно прозрачным способом. Применение нечётких систем для решения проблем автоматического управления, прогнозирования, распознавания образов, принятия решений заставляет специалистов искать эффективные методы построения таких систем, для идентификации которых наряду с алгоритмами оптимизации, основанными на производных, используются метаэвристики — чаще всего генетические алгоритмы и нейронные сети. В последнее время для этих целей стали применять алгоритмы муравьиной колонии (АМК). Наиболее популярен данный алгоритм при настройке нечётких правил нечёткого регулятора [1–5]. В работе [2] antecedentes правил задаются априорно, консеквенты назначаются на основе метода Q -обучения и АМК. В [3] выбор консеквентов ведётся также на основе двух подходов: методологии COR (Cooperative Rules), устанавливающей сотрудничество нечётких правил, и АМК. Параметры нечёткого контроллера в работе [4] настраиваются с помощью метода нечёткой муравьиной колонии со специфическим нечётко-вероятностным выбором следующего узла графа. Для формирования antecedentes правил в [5] используется нечёткий кластерный анализ, а консеквенты формируются на основе АМК. В работе [6] решается задача структурной идентификации нечёткой системы одновременно с установкой консеквентов правил. Antecedents правил при этом не настраиваются. Собственно задача идентификации нечёткой системы сведена здесь к решению квадратичной задачи назначений, и для её решения используется дискретный АМК. Совместное применение АМК и градиентного метода для идентификации нечётких систем представлено в [7]. Все перечисленные приложения основаны на традиционном АМК, ориентированном на решение комбинаторных проблем, связанных с поиском оптимальных путей на графе [8]. Однако идентификация

*Работа выполнена при поддержке Российского фонда фундаментальных исследований (проект № 09-07-99008).

параметров нечётких систем относится к задачам непрерывной оптимизации, и применение классического алгоритма в силу его изначальной дискретности не всегда позволяет достичь заданной точности. В работе [9] представлен непрерывный алгоритм муравьиной колонии ACO_R .

Целью предлагаемой работы является описание применения ACO_R для идентификации нечётких систем, которое увеличивает точность нечёткого вывода.

Постановка задачи. Рассматривается нечёткая система типа синглтон, i -е правило такой системы имеет следующий вид:

$$\text{IF } x_1 = A_{1i} \text{ AND } x_2 = A_{2i} \text{ AND } \dots \text{ AND } x_n = A_{ni} \text{ THEN } y = r_i,$$

где A_{ij} — лингвистический терм для оценивания переменной x_i ; r_i — значение консеквента i -го правила.

Нечёткая система осуществляет отображение $f: \mathfrak{R}^n \rightarrow \mathfrak{R}$:

$$f(\mathbf{x}) = \sum_{i=1}^R \mu_{A_{1i}}(x_1) \mu_{A_{2i}}(x_2) \dots \mu_{A_{ni}}(x_n) r_i / \sum_{i=1}^R \mu_{A_{1i}}(x_1) \mu_{A_{2i}}(x_2) \dots \mu_{A_{ni}}(x_n).$$

Здесь \mathbf{x} — входной вектор; R — число правил; n — количество входных переменных; $\mu_{A_{ij}}$ — функция принадлежности (ФП).

Нечёткая система может быть представлена как

$$y = f(\mathbf{x}, \boldsymbol{\theta}, \mathbf{r}),$$

где $\boldsymbol{\theta} = \|\theta_1, \dots, \theta_N\|$ — вектор параметров antecedentes, N — число идентифицируемых параметров ($N = n \times (\text{число параметров, описывающих одну ФП}) \times (\text{число термов, описывающих одну входную переменную})$); y — скалярный выход системы; $\mathbf{r} = \|r_1, \dots, r_R\|$ — вектор параметров консеквентов.

Пусть дано множество обучающих данных (таблица наблюдений) $\{(\mathbf{x}_p; t_p), p = 1, \dots, m\}$, тогда среднеквадратическая функция ошибки, являющаяся численным критерием адекватности модели, вычисляется по формуле

$$E(\mathbf{x}_p, \boldsymbol{\theta}, \mathbf{r}) = \sqrt{\sum_{p=1}^m (t_p - f(\mathbf{x}_p, \boldsymbol{\theta}, \mathbf{r}))^2 / m}.$$

Проблема идентификации сводится к проблеме поиска минимума заданной функции в многомерном пространстве, координаты которого соответствуют параметрам нечёткой системы. В силу того что поверхность поиска в указанном пространстве имеет сложный рельеф, методы поиска, основанные только на производных, здесь не всегда эффективны. Для решения проблемы минимизации параметров antecedentes предлагается использовать непрерывный алгоритм муравьиной колонии (ACO_R), а для настройки консеквентов — метод наименьших квадратов (МНК) [10].

Непрерывный алгоритм муравьиной колонии. В классическом алгоритме муравьиной колонии при выборе дуги муравей «руководствуется» дискретным распределением вероятности. В случае непрерывного алгоритма этот выбор не ограничен конечным множеством, здесь дискретное распределение заменяется непрерывным, заданным своей функцией плотности [9]. В алгоритме ACO_R используется функция плотности вероятности с гауссовым ядром. Гауссово ядро $G^i(x)$ в работе [9] основано на взвешенной сумме нескольких одномерных гауссовых функций $g_l^i(x)$:

$$G^i(x) = \sum_{l=1}^k \omega_l g_l^i(x) = \sum_{l=1}^k \omega_l \frac{1}{\sigma_l^i \sqrt{2\pi}} e^{-(x - \theta_l^i)^2 / (2\sigma_l^{i2})}. \quad (1)$$

Каждому параметру нечёткой модели соответствует своё гауссово ядро, $i = 1, \dots, N$. Каждая функция $G^i(x)$ описывается тремя векторами: $\theta^i = \{\theta_1^i, \theta_2^i, \dots, \theta_k^i\}$, ω — вектор весов, связанных с индивидуальными гауссовыми функциями, σ^i — вектор среднеквадратичных отклонений, который вычисляется следующим образом:

$$\sigma_l^i = \xi \sum_{j=1}^k |\theta_j^i - \theta_l^i| / (k - 1).$$

Параметр $\xi > 0$ является одинаковым для всех размерностей и имеет эффект, подобный норме испарения феромона в дискретном алгоритме муравьиной колонии. Чем больше значение ξ , тем ниже скорость сходимости алгоритма [9].

В непрерывном алгоритме вводится понятие архива решений, который представлен таблицей из k строк. Каждая строка состоит из трёх частей:

- 1) найденное муравьём решение $\theta_l = \{\theta_l^1, \theta_l^2, \dots, \theta_l^N\}$;
- 2) ошибка, вычисленная нечёткой системой;
- 3) вес решения ω_l .

Решения упорядочены в архиве по возрастанию ошибки. Вес ω_l решения θ_l определяется согласно формуле

$$\omega_l = \frac{1}{qk\sqrt{2\pi}} e^{-(l-1)^2/(2q^2k^2)},$$

где q — задаваемый параметр алгоритма.

При добавлении нового решения в архив худшее из них удаляется. Этот процесс аналогичен процессу испарения феромона в классическом АМК.

Муравьи в алгоритме делятся на колонии, каждая из которых отвечает за нахождение параметров своей ФП. У каждой колонии свой независимый архив решений.

Рассмотрим собственно алгоритм для идентификации параметров нечёткой модели:

Шаг 1. Задать начальные параметры.

Шаг 2. Сгенерировать популяцию муравьёв в колониях.

Шаг 3. Сгенерировать несколько случайных решений для всех архивов решений с последующим оцениванием и ранжированием.

Шаг 4. Найти значения вектора весов. Сделать текущим первого муравья первой колонии.

Шаг 5. Для текущего муравья текущей колонии вычислить номер используемой функции Гаусса. Определить параметры функций Гаусса. Сгенерировать N случайных величин $\{\theta_1, \dots, \theta_N\}$ на основе полученных функций.

Шаг 6. Найти ошибку вывода нечёткой системы при параметрах $\{\theta_1, \dots, \theta_N\}$; если ошибка меньше текущей, то сохранить новые параметры.

Шаг 7. Добавить в архив новое решение, ранжировать архив, удалить из архива худшее решение.

Шаг 8. Если в текущей колонии имеется следующий муравей, то сделать его текущим и перейти к шагу 5, иначе перейти на шаг 9.

Шаг 9. Если имеется следующая колония, то сделать текущим первого муравья в этой колонии и перейти на шаг 5, иначе перейти на шаг 10.

Шаг 10. Если условие окончания работы алгоритма выполнено, то закончить, иначе сделать текущим первого муравья первой колонии и перейти к шагу 5.

Условием окончания работы алгоритма является достижение определённого числа итераций либо получение ошибки меньше заданной.

Исследование алгоритма идентификации нечётких систем проводилось при решении задач аппроксимации. В качестве тестовых были выбраны следующие функции:

$$1) f(x_1, x_2) = x_1 \sin(x_2), \quad -\pi/2 < x_1, \quad x_2 < \pi/2;$$

$$2) f(x_1, x_2, x_3) = 1 + x_1^{0,5} + x_2^{-1} + x_3^{-1,5}, \quad x_1, x_2, x_3 \in [1, 5];$$

$$3) f(x) = (1 + 10 \exp(-100(x - 0,7)^2)) \left(\frac{\sin(125/(x + 1,5))}{x + 0,1} \right), \quad 0 < x < 1;$$

$$4) f(x_1, x_2) = (1 + x_1^{-2} + x_2^{-1,5})^2, \quad x_1, x_2 \in [1, 5].$$

На основе тестовой функции формировалась таблица наблюдений, по которой строилась нечёткая система, аппроксимирующая данную функцию. Критерием качества аппроксимации была среднеквадратическая ошибка вывода.

В процессе исследования непрерывного алгоритма изменялись следующие его параметры: число итераций, размер архива решений, число муравьёв, параметр q .

Алгоритм имеет свойство застревать в локальных минимумах, по этой причине необходимо говорить о лучших и худших решениях при одних и тех же параметрах алгоритма. Разные решения получаются в силу случайности, присутствующей в самом алгоритме, однако не всегда этот случайный выбор способен вывести алгоритм из локального минимума.

Распределения среднеквадратической ошибки от числа итераций (число испытаний 10) приведены на рис. 1. Эксперимент показал, что точность решения зависит от числа итераций в определённых пределах. Особенно ярко это проявляется для функции 3. Лучшие решения (малые ошибки) появляются при числе итераций 80 для функций 1–3 и при 40 итерациях для функции 4.

Распределения среднеквадратической ошибки от размера архива решений (число испытаний 10) показаны на рис. 2. Диапазон лучших значений размера архива решений для аппроксимации различных функций следующий: 20–40 для функции 1, 30–180 для функции 3. Для функций 2 и 4 диапазон оптимальных значений менее выражен.

Распределения среднеквадратической ошибки от числа муравьёв (число испытаний 10) представлены на рис. 3. В процессе эксперимента выявлено, что точность решения зависит от числа муравьёв только при аппроксимации функции 3, для остальных функций такая зависимость проявляется не столь очевидно.

В ходе эксперимента установлено, что точность решения не зависит от значения константы q . Лучшие решения (малые ошибки) появляются как при значении константы $q = 0,05$, так и при $q = 1$.

Компьютерный эксперимент позволил установить, что зависимость времени вывода от числа итераций и числа муравьёв носит линейный характер. В табл. 1 приведены некоторые регрессионные модели зависимости времени вывода от числа итераций и числа муравьёв, в скобках указаны квадраты коэффициентов корреляции R^2 как мера точности полученных зависимостей. Приведённые регрессионные уравнения увеличивают информативность экспериментальных результатов и позволяют установить тенденцию изменения времени вывода в зависимости от числа итераций работы алгоритма и числа муравьёв.

Сравнение разработанного алгоритма с аналогами. Для сравнения разработанного алгоритма с существующими подходами построения нечётких моделей было проведено исследование результатов аппроксимации нескольких нелинейных функций. В силу того что непрерывный АМК в представленной реализации оптимизирует только antecedentes правил, нами был включён в цикл идентификации МНК для настройки консеквентов правил [10].

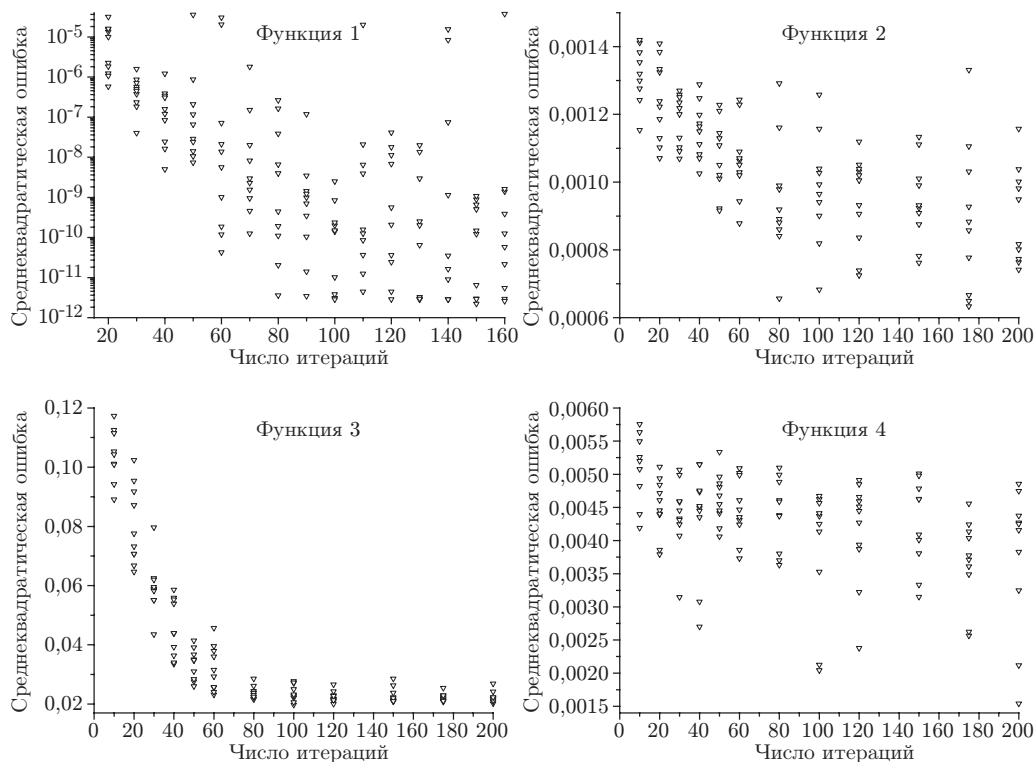


Рис. 1. Зависимость среднеквадратической ошибки от числа итераций (25 муравьёв, 100 — архив, $q = 0,8$)

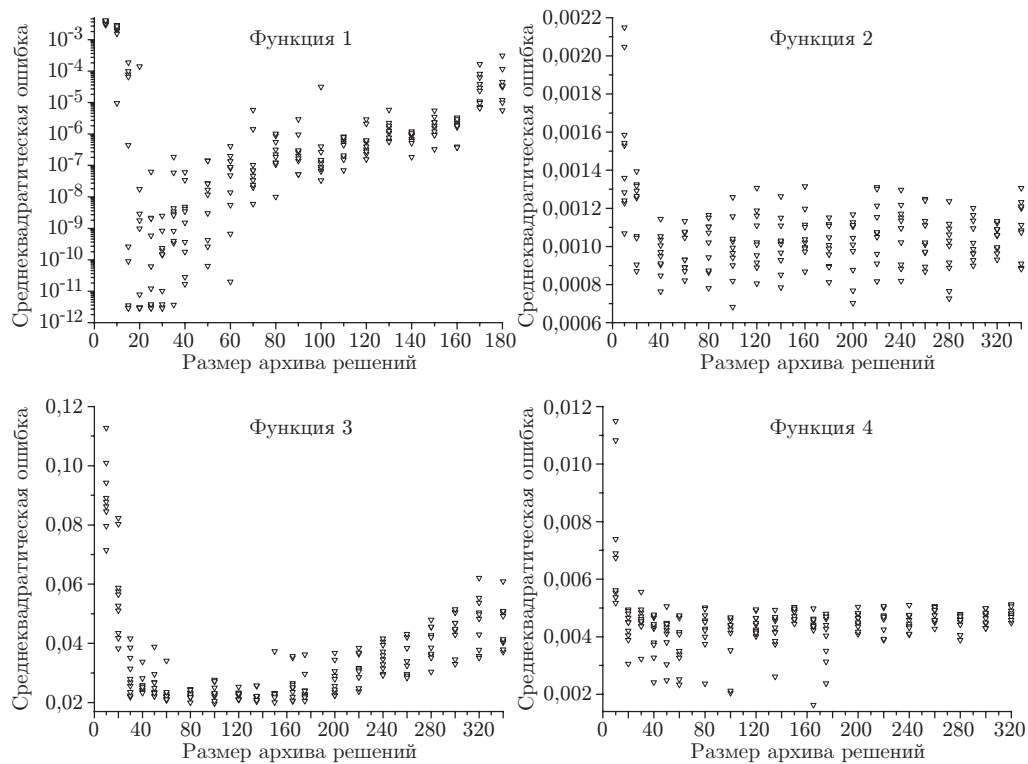


Рис. 2. Зависимость среднеквадратической ошибки от размера архива решений (25 муравьёв, 100 итераций, $q = 0,8$)

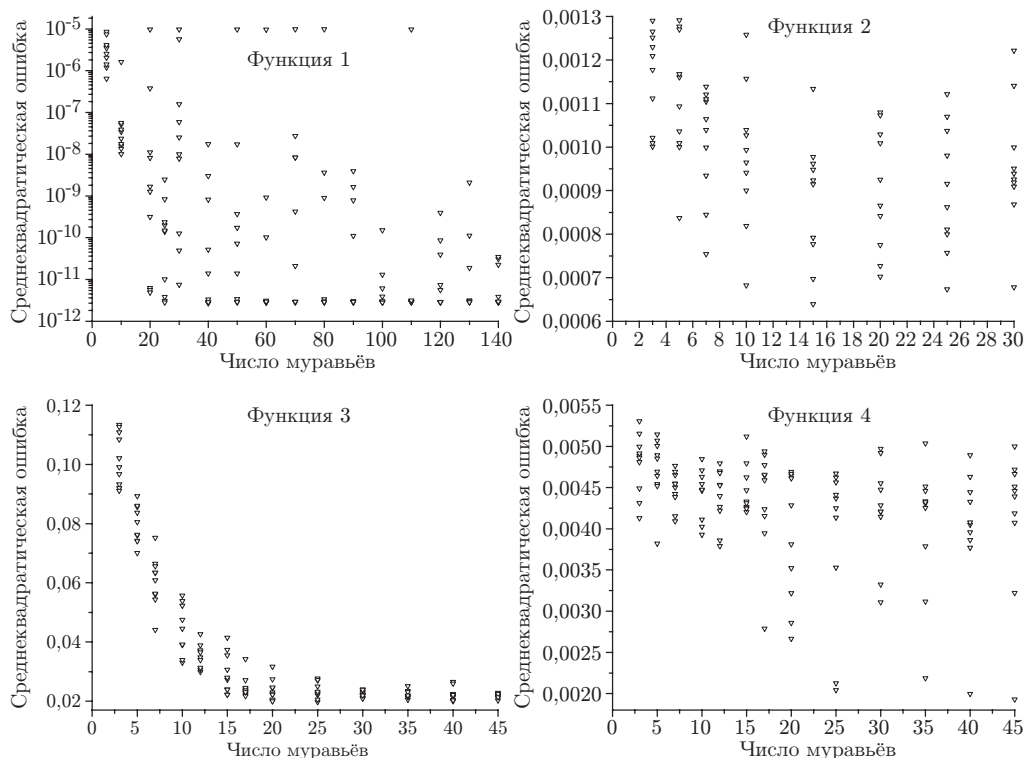


Рис. 3. Зависимость среднеквадратической ошибки от числа муравьёв (100 — архив, 100 итераций, $q = 0,8$)

Таблица 1

Зависимость времени идентификации от числа итераций и числа муравьёв

Функция	Число входных переменных	Число точек в таблице наблюдений	Число термов, представляющих входную переменную	Зависимость времени от числа итераций	Зависимость времени от числа муравьёв
1	2	81	5	$-0,74881 + 0,145357 \text{iter}$ ($R^2 = 0,997085$)	$-0,870192 + 0,584188 \text{ant}$ ($R^2 = 0,999237$)
4	2	121	5	$-2,28372 + 0,241937 \text{iter}$ ($R^2 = 0,998553$)	$-2,97005 + 1,01452 \text{ant}$ ($R^2 = 0,998664$)
2	3	216	4	$-6,92796 + 0,472145 \text{iter}$ ($R^2 = 0,997546$)	$-6,94595 + 4,52162 \text{ant}$ ($R^2 = 0,996243$)

	ФП	1	2	3
▶	1	0	0	0,04691844
	2	0,00519209	0,05311402	0,11410635
	3	0,06645587	0,11516812	0,18667138
	4	0,13397047	0,18300861	0,25649313
	5	0,19715601	0,26295571	0,33744215
	6	0,27735528	0,34886256	0,42255385

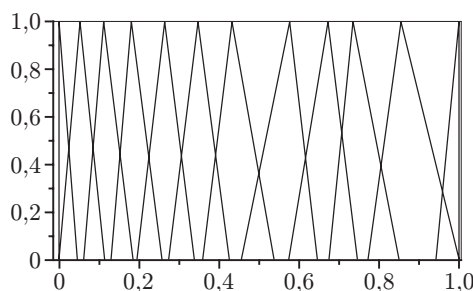


Рис. 4. Функции принадлежности для аппроксимации функции 3

Аппроксимация тестовой функции 3 проводилась на 12 правилах как в данной работе, так и в аналогичных. В [11] для алгоритма Mitaim — Kosko ошибка аппроксимации составляла 1,426, для алгоритма Lisin — Gennert ошибка 0,247, для непрерывного АМК + МНК ошибка 0,0191.

Значения параметров шести первых ФП и их общий вид представлены на рис. 4.

База правил нечёткой системы, аппроксимирующей функцию 3, приведена в табл. 2. Тройки параметров, задающих ФП, указаны на рис. 4.

Настройка нечётких моделей разработанным алгоритмом и аналогами для аппроксимации тестовой функции 4 производилась по таблице наблюдений, состоящей из 400 строк. Результаты работы АСО_R + МНК и алгоритмов, описанных в [12], приведены в табл. 3. Как видно из таблицы, предлагаемый алгоритм превосходит аналоги.

Вид и значения параметров ФП нечёткой системы, аппроксимирующей тестовую функцию 4, представлены на рис. 5.

Таблица 2

База правил для аппроксимации функции 3

	ФП		r
	1		9,919
	2		-6,175
	3		4,372
	4		-3,680
	5		2,663
IF $x =$	6	THEN $y =$	-2,227
	7		1,742
	8		-2,789
	9		11,851
	10		-8,656
	11		0,938
	12		-0,103

Таблица 3

Результаты аппроксимации функции 4

Алгоритм	Количество правил	Среднеквадратическая ошибка
Rojas, Pomares, Ortega, Prieto	9	0,146
	16	0,051
	25	0,026
	36	0,017
Teng, Wang, Chiu	4	0,016
Непрерывный АМК + МНК	9	0,00538
	16	0,00215
	25	0,00186
	36	0,00129

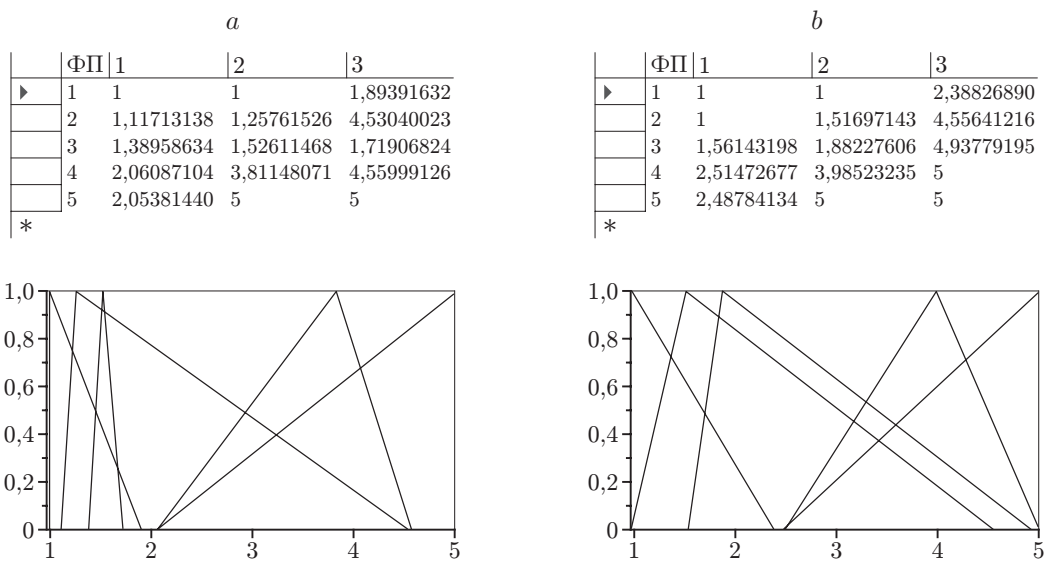


Рис. 5. Функции принадлежности для аппроксимации функции 4: *a* — для переменной x_1 , *b* — для переменной x_2

При аппроксимации тестовой функции 2 идентификация нечёткой системы разработанным алгоритмом и аналогом проводилась на таблице наблюдений, состоящей из 216 строк. В работе [13] аппроксимация выполнялась адаптивным алгоритмом роящихся частиц с 9 правилами, ошибка составила 0,00243. Гибридный алгоритм АСО_R + МНК на 27 правилах уменьшает ошибку в среднем до 0,00136.

Заключение. В данной работе впервые для идентификации нечётких моделей использован непрерывный алгоритм муравьиной колонии. Имитационные эксперименты и сравнительный анализ с аналогами показали, что у предложенного алгоритма есть преимущества перед другими методами идентификации параметров нечётких систем, решающих задачи аппроксимации.

СПИСОК ЛИТЕРАТУРЫ

1. **Востриков А. С.** Проблема синтеза регуляторов для систем автоматике: состояние и перспективы // Автометрия. 2010. **46**, № 2. С. 3–19.
2. **Juang C.-F., Lu C.-M.** Ant colony optimization incorporated with fuzzy Q-learning for reinforcement fuzzy control // IEEE Trans. Systems, Man, and Cybernetics. Pt. A. 2009. **39**, N 3. P. 597–608.
3. **Mucientes M., Casillas J.** Quick design of fuzzy controllers with good interpretability in mobile robotics // IEEE Trans. Fuzzy Systems. 2007. **15**, N 4. P. 636–651.
4. **Tao C.-W., Taur J.-S., Jeng J.-T., Wang W.-Y.** A novel fuzzy ant colony system for parameter determination of fuzzy controllers // Intern. Journ. Fuzzy Systems. 2009. **11**, N 4. P. 298–307.
5. **Juang C.-F., Lo C.** Fuzzy systems design by clustering-aided ant colony optimization for plant control // Intern. Journ. General Systems. 2007. **36**, N 6. P. 623–641.
6. **Casillas J., Cordon O., Fernandez de Viana I., Herrera F.** Learning cooperative linguistic fuzzy rules using the best-worst ant system algorithm // Intern. Journ. Intelligent Systems. 2005. **20**, N 4. P. 433–452.
7. **Ходашинский И. А., Дудин П. А.** Параметрическая идентификация нечетких моделей на основе гибридного алгоритма муравьиной колонии // Автометрия. 2008. **44**, № 5. С. 24–35.
8. **Dorigo M., Maniezzo V., Coloni A.** Ant system: optimization by colony of cooperating agents // IEEE Trans. Systems, Man, and Cybernetics. Pt. B. 1996. **26**, N 1. P. 29–41.
9. **Socha K., Dorigo M.** Ant colony optimization for continuous domains // Europ. Journ. Operational Research. 2008. **185**, N 3. P. 1155–1173.
10. **Ходашинский И. А., Гнездилова В. Ю., Дудин П. А., Лавыгина А. В.** Основанные на производных и метаэвристические методы идентификации параметров нечетких моделей // Тр. VIII Междунар. конф. «Идентификация систем и задачи управления» (SICPRO'08). М.: Институт проблем управления им. В. А. Трапезникова РАН, 2009. С. 501–529.
11. **Lisin D., Gennert M. A.** Optimal function approximation using fuzzy rules // Proc. of the 18th Intern. Conf. North American Fuzzy Information Processing Society. N. Y.: IEEE Press, 1999. P. 184–188.
12. **Teng Y.-W., Wang W.-J., Chiu C.-H.** Function approximation via particular input space partition and region-based exponential membership functions // Fuzzy Sets and Systems. 2004. **142**, N 2. P. 267–291.
13. **Aliyari M., Teshnehlab M. Sh., Sedigh A. K.** Novel hybrid learning algorithms for tuning ANFIS parameters using adaptive weighted PSO // IEEE Intern. Conf. on Fuzzy Systems. London: Imperial College, 2007. P. 1–6.

Поступила в редакцию 12 апреля 2011 г.
