

УДК 004.89:519.6

## ДИСКРЕТИЗАЦИЯ ПОЛОЙ ТРЕХМЕРНОЙ ОБЛАСТИ НЕЙРОСЕТЕВЫМИ МЕТОДАМИ

**Архипов С.В., Цыренов Ц.В.**

*ФГБОУ ВО «Бурятский государственный университет»,  
Улан-Удэ, e-mail: svarkh@bsu.ru, tsyrya@gmail.com*

Работа посвящена проблеме дискретизации полой пространственной области на основе нейросетевой модели самоорганизующихся карт Кохонена (SOM). Приведен модифицированный алгоритм SOM и его программная реализация построения адаптивной сетки для полой пространственной области. Обсуждены особенности адаптации внутренней границы области и выбора параметров обучения. В алгоритме выделены две макроитерации, соответствующие этапам уточнения и упорядочивания. Для получения лучшего результата в модифицированном алгоритме SOM предлагается в качестве параметра Гауссовой функции расстояния использовать линейно убывающую функцию на этапе уточнения и экспоненциально убывающую в последующем. Исследована зависимость экстремальных значений параметра гауссовой функции от числа итераций на каждом из этапов обучения нейросети. Рекомендации по выбору параметров обучения хорошо согласуются с результатами построения адаптивных сеток для плоских односвязных областей.

**Ключевые слова:** дискретизация, адаптивные сетки, полая трехмерная область, нейросетевые алгоритмы, самоорганизующиеся карты SOM

## SAMPLING OF HOLLOW SPATIAL DOMAIN BY NETWORK MEANS

**Arkhipov S.V., Tsyrenov Ts.V.**

*Federal Government Budget Educational Institution of Higher Education Buryat State University,  
Ulan-Ude, e-mail: svarkh@bsu.ru, tsyrya@gmail.com*

The paper deals with the problem of sampling of hollow spatial domain on the basis of neural network model of Kohonen Self-Organizing Maps (SOM). The modified algorithm of SOM and its program implementation of the adaptive grid for hollow spatial domain are presented here. This paper analyses adaptation trends of domain inner boundary and parameters selection for training. There are two macro iterations corresponding to stages of specification and regulation in the algorithm. For best results it is offered to use linearly decreasing function at a stage of specification and then exponentially decreasing function as the parameter of Gaussian function in the modified algorithm of SOM. The dependence of the parameter extremal values from the number of iterations at each stage of neural network training is studied. The recommendations on selection of training parameters are in accord with the results of the adaptive grids building for plane simply connected domains.

**Keywords:** sampling, adaptive grids, hollow three-dimensional domain, neural network algorithms, self-organizing maps (SOM)

Адаптивные сетки, представляя дискретную модель физической области, нашли важное применение при решении сложных задач численного моделирования во многих областях науки, таких как гидродинамика, динамика твердого тела, теория упругости, термодинамика и т.д. [2, 3, 6, 7, 13].

Интенсивное развитие вычислительной техники значительно расширило возможности численных алгоритмов решения прикладных задач. Известны многочисленные теоретические и экспериментальные результаты, показывающие преимущества использования адаптивных сеток при решении сложных многомерных задач [12] и др.

Традиционными методами построения адаптивных сеток являются метод эквираспределения [6], Томпсона [13], эллиптический метод [3], алгебраические методы [7], конформных отображений [2] и т.д. В основе большинства применяемых методов лежат

теории дифференциальных уравнений, вариационного исчисления и дифференциальной геометрии. При этом построение адаптивных сеток требует решения сложных систем нелинейных дифференциальных уравнений с частными производными, что накладывает ряд известных ограничений.

В настоящее время высокую эффективность построения адаптивных сеток с заданной плотностью на сложной физической области демонстрируют нейросетевые алгоритмы [4, 5, 8, 11] и т.д. Развитие современных нейросетевых моделей обязано классической теории самоорганизующихся карт Кохонена (SOM – Self-Organizing Maps) (например, [9, 10]). Этот метод обладает рядом преимуществ по сравнению с классическими методами, он прост в реализации и может использовать произвольные начальные данные, например равномерные треугольные, прямоугольные или

шестиугольные сетки. Данный метод не зависит от размерностей вычислительного или физического пространств. Это значит, что с помощью этого метода можно строить адаптивные сетки любой размерности на одно-, двух- и трехмерных областях.

Наиболее важными свойствами модели SOM являются, во-первых, ее способность строить отображения произвольной размерности на основе самоорганизации и методов теории вероятностей и, во-вторых, наличие у нее внутреннего параллелизма, как у представителя нейронных сетей. Однако при использовании базовой модели SOM возникает ряд проблем, упоминающихся во многих источниках, например [5], к числу которых относятся граничный эффект, наличие «мертвых» нейронов и нарушение сохранения топологии при отображении. Эти проблемы ограничивают возможности применения модели SOM, в том числе и для построения адаптивных сеток. Было установлено, что граничный эффект приводит к неплотному прилеганию сетки к границе физической области, «мертвые» нейроны появляются при выходе узлов сетки за границу области, а нарушение сохранения топологии приводит к вырожденности сетки.

Для решения указанных проблем разработаны модифицированные методы, в основе которых лежат идеи чередования базового алгоритма для внутренних и внешних узлов сетки [11], а также усовершенствования функции соседства нейронов с усложнением параметров функций расстояния и скорости обучения относительно дискретного времени [4, 5].

Настоящая работа посвящена совершенствованию нейросетевых алгоритмов построения адаптивных сеток на сложных пространственных областях, характеризующихся пористой структурой, программой реализации и апробации предложенных алгоритмов, выработке рекомендаций по выбору параметров обучения. В основе применяемых алгоритмов лежит идея чередования базового алгоритма для граничных и внутренних узлов сетки, а также использования упрощенных формул функции соседства нейронов. Вычислительные эксперименты показали эффективность рекомендаций по выбору параметров обучения нейросети на односвязной плоской области [1] и в случае трехмерной полый области.

### Основные идеи

Как известно, нейросетевая модель SOM состоит из карты нейронов, опре-

деленным образом расположенных друг относительно друга в пространстве, которое называется выходным. Наиболее распространенным расположением нейронов в карте является их расположение в узлах прямоугольной сетки. С каждым нейроном связан весовой вектор, являющийся элементом пространства, которое называется входным пространством. Весовой вектор корректируется в процессе итерационного алгоритма обучения модели SOM, используя элементы обучающего множества, состоящего из конечного числа векторов входного пространства. Считается, что обучающее множество – это выборка из вероятностного распределения, которое требуется аппроксимировать весовыми векторами нейронов в результате обучения.

Рассмотрим алгоритм модифицированной модели SOM для пространственной адаптивной сетки на многосвязной пространственной области. Под многосвязной областью будем подразумевать область  $G$ , для которой существует замкнутая поверхность, не стягиваемая в пределах области непрерывной деформацией в точку.

Пусть в области  $G$  выделены внутренние границы  $G_1, G_2, \dots, G_{n-1}$ , внешняя граница  $G_n$  и внутренность  $G_{n+1}$ . Такое разбиение позволит использовать базовую модель SOM на всей области  $G$ , на ее внешней  $G_n$  и внутренних  $G_1, G_2, \dots, G_{n-1}$  границах, а также внутренности  $G_{n+1}$ . Пусть  $W_{ijk} = N \times N \times N$  – это равномерная кубическая сетка, зафиксированная на области  $G$ . Подмножество индексов граничных узлов  $W_n$  описывает узлы  $G_n$ , т.е. узлы, расположенные на границе области  $G$ ,  $W_{n+1}$  описывает  $G_{n+1}$  – внутренние узлы  $G$ , а  $W_1, W_2, \dots, W_{n-1}$  – узлы, расположенные на соответствующих внутренних границах  $G_1, G_2, \dots, G_{n-1}$ .

В этих обозначениях модифицированный алгоритм можно описать следующей последовательностью шагов.

1. Устанавливаются начальные веса  $w_{ijk}$  всех нейронов в вершинах трехмерной сетки.

2. На первой макроитерации ( $s = 1$ ), соответствующей дискретному времени  $t \in [1, t_0]$ , выполняется:

2.1. Генерируется случайная точка  $x(t)$  во всей области  $G$ ;

2.2. Определяется нейрон победитель в евклидовой метрике  $d$ . Фиксируются сеточные координаты нейрона победителя  $i_{BMU}, j_{BMU}, k_{BMU}$  ( $BMU$  – best matching unit) для образца  $x(t)$ ;

2.3. Настраиваются новые весовые значения нейронов сети по формуле

$$w_{ijk}(t+1) = w_{ijk}(t) + \theta(t, i_{BMU}, j_{BMU}, k_{BMU}, i, j, k)(x(t) - w_{ijk}(t)), \quad (1)$$

где

$$\theta(t, i_{BMU}, j_{BMU}, k_{BMU}, i, j, k) = \delta(t) h(d, t) \quad (2)$$

– функция соседства нейронов;  $t$  – номер итерации;  $\delta(t)$  – функция скорости обучения;  $h(d, t)$  – функция расстояния.

3. Осуществление вырезов:

3.1. Удаление нейронов, попавших в полость области  $G$ .

3.2. Определение внутренних границ сетки  $W_1, W_2, \dots, W_{n-1}$ .

4. На каждой макроитерации  $s > 1$  выполняется:

4.1. В течение  $T_1(s)$  итераций:

4.1.1. Генерируется случайная точка  $x(t)$  на одной из границ (внешней/внутренней) области  $G$ , т.е. точка принадлежащая одной из областей  $G_1, G_2, \dots, G_n$ ;

положения сетки в области. Последующие макроитерации уточняют расположение сетки относительно границы и внутренней области  $G$ .

В работе рассмотрена упрощенная функция соседства (2). Так за скорость обучения выбрана убывающая степенная функция  $\delta(t) = t^{-0.2}$ , за функцию расстояния выбрана функция Гаусса

$$h(d, t) = e^{-\frac{d^2}{2\sigma^2(t)}},$$

где  $\sigma(t) = a \cdot t^{-0.2}$  – радиус обучения, параметр  $a$ :

$$a(t) = \begin{cases} \frac{a(t_0) - a(1)}{t_0}(t_0 - t) + a(1), & \text{при } 1 \leq t \leq t_0; \\ a(t_0) \left( 1 - e^{-\frac{5(t - t_{\max} - t_0)}{t_{\max} + t_0}} \right) (0,005)^{\frac{t - t_0}{t_{\max} - t_0}} + a_{\min}, & \text{при } t_0 < t \leq t_{\max}. \end{cases} \quad (3)$$

4.1.2. Определяется нейрон победитель  $BMU$  из граничных узлов сетки, индексы которых входят в  $W_i$ , где  $i$  – индекс области  $G_i$ , из которой был выбран случайный объект;

4.1.3. Настраиваются новые весовые значения граничных узлов сетки, индексы которых входят в  $W_i$ , по формуле (1).

4.2. В течение  $T_2(s)$  итераций:

4.2.1. Генерируется случайная точка  $x(t)$  во всей области  $G$ ;

4.2.2. Определяется нейрон победитель  $BMU$  среди всех узлов сетки;

4.2.3. Если в шаге 4.2.2 победил нейрон сетки, индексы которого входят в  $W_i$ ,  $i = 1, n$ , то случайно сгенерированная точка  $x(t)$  заменяется на этот нейрон;

4.2.4. Настраиваются новые весовые значения внутренних узлов сетки, индексы которых входят в  $W_{n+1}$ , по формуле (1).

5. Повторяются макроитерации до тех пор, пока изменения положений узлов не станут достаточно малыми.

Первая макроитерация в классической терминологии соответствует этапу упорядочивания. Здесь важным показателем является правильность предварительного рас-

следует отметить, что использование формулы (3) в качестве параметра  $a(t)$  продиктовано работой [1].

В представленном алгоритме наибольшее время занимает процесс нахождения ближайшего узла сетки к заданной точке пространства. Поскольку поиск  $BMU$  применяется не один раз, то минимизация времени и числа операций является важной подзадачей. Чтобы ускорить вычисления, производится дихотомия над узлами сетки – разбиение области сетки на одинаковые элементарные ее подобласти – кубики. При этом сложность рекурсивного поиска можно оценить как  $O(\log)$ , когда как сложность простого перебора узлов сетки  $N \times N \times N$  равна  $O(N^3)$ .

Еще одним относительно трудоемким процессом при построении сетки является процесс корректировки узлов. Так как сдвиг узла определяется по формуле, которая не зависит от других узлов сетки, то эти операции можно выполнять параллельно. Поэтому при реализации алгоритма операции корректировки узлов целесообразно распараллелить на  $N^3$  процессов.

### Вычислительные эксперименты

Вычислительные эксперименты проводились на симметричной выпуклой области  $G$  – куб с полостью в форме сферы. Выбор симметричной формы для вычислительных экспериментов обусловлен очевидностью выбора правильного расположения адаптивной сетки в процессе вариаций параметров обучения.

При анализе качества построения сетки в модифицированном алгоритме целесообразно разделить этапы упорядочивания и уточнения. В следующих вычислительных экспериментах исследовалось качество построения сетки при вариациях параметров  $a$  и  $t_0$ . За основу на первом этапе обучения принята [1] линейная зависимость параметра  $a$  от дискретного времени  $t$ .

На рис. 1 приведены итоги стадии упорядочивания в зависимости от числа итера-

ций  $t_0$ . Как видно (рис. 1, в), с увеличением числа итераций сеть, приспосабливаясь к особенностям области, на 100000 итерации приняла оптимальное положение. Под оптимальным положением понимается правильная ориентация сетки – для квадратной области в соответствии с расположением вершин квадрата.

Уменьшение нижнего предела параметра  $a$  на этапе упорядочивания при неизменных  $t_0$  и  $a(1)$  приводит к ухудшению качества предварительного построения сетки (рис. 2). Это объясняется тем, что с уменьшением  $a(t_0)$  уменьшается и радиус обучения  $\sigma(t)$ , а следовательно, основное смещение узлов сетки происходит в окрестности сгенерированных точек. В результате нарушается не только гладкость сетки, но и увеличивается ее размер, что значительно ухудшает качество адаптации на втором этапе обучения.

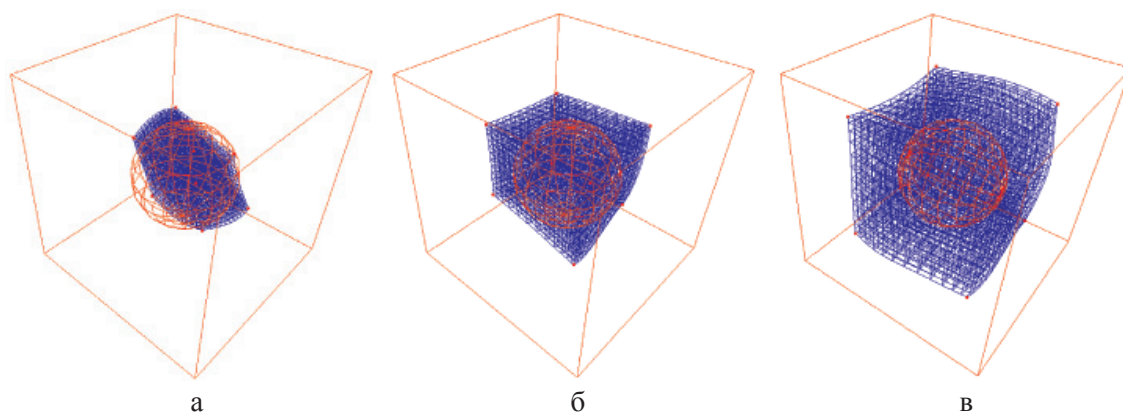


Рис. 1. Предварительное расположение сетки в области при  $a(1) = 100$ ,  $a(t_0) = 25$  по истечении  $t_0$  итераций:  
 $a - t_0 = 1000$ ;  $б - t_0 = 5000$ ;  $в - t_0 = 100000$

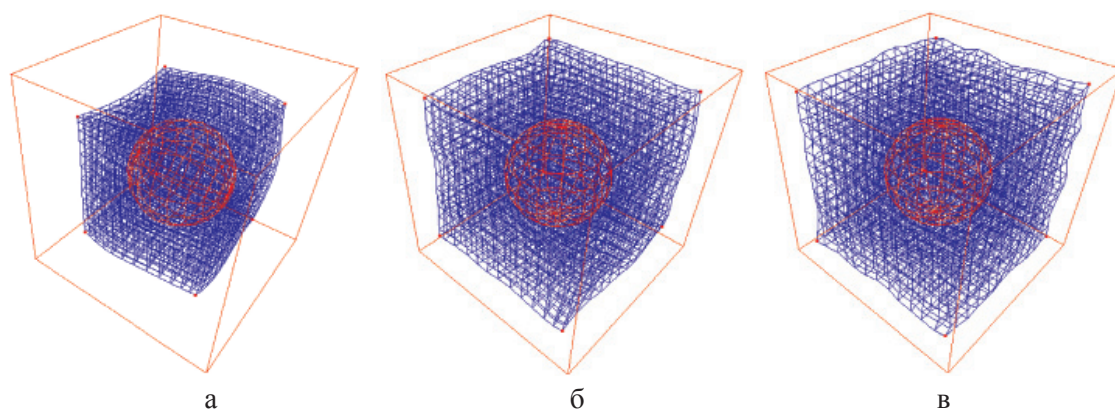


Рис. 2. Предварительное расположение сетки в области при  $a(1) = 100$ ,  $t_0 = 100000$  и различных значений параметра  $a$ :  
 $a - a(t_0) = 25$ ;  $б - a(t_0) = 10$ ;  $в - a(t_0) = 5$

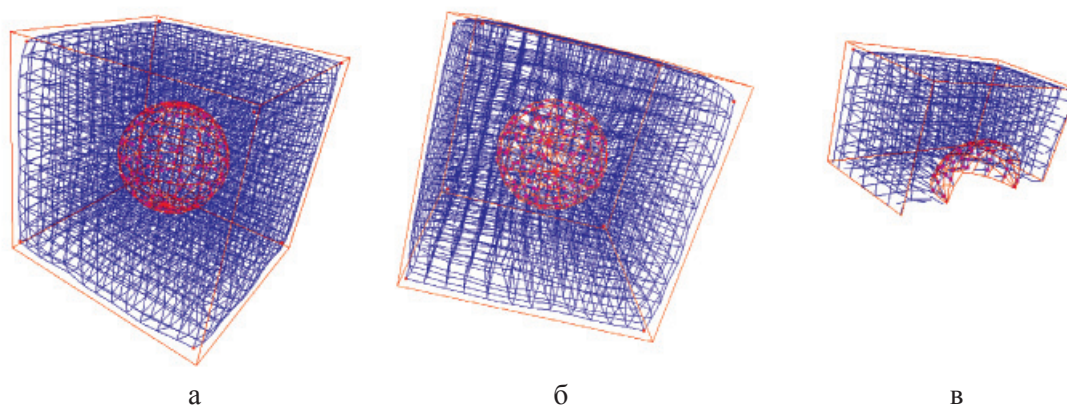


Рис. 3. Результат применения алгоритма для полой трехмерной области:  
а – вид сбоку; б – вид снизу; в – в разрезе

Исследовалась зависимость качества построения адаптивной сетки от верхнего предела параметра  $a$  при неизменных  $t_0$  и  $a(t_0)$ . Выявлено, что при уменьшении верхнего предела параметра  $a$  качество адаптации сетки значительно снижается. Вычисления показывают, что при достаточно большом радиусе начального обучения, например при  $a(1) = 100$ , в первых итерациях сеть «сжимается» к случайно сгенерированным точкам, уменьшаясь в размерах. После чего медленно разворачивается, адаптируясь к особенностям области. При уменьшении начального радиуса сеть минуется стадию «сжатия», и начальные размеры сети уменьшают возможности правильной адаптации.

На стадии уточнения эксперименты подтвердили [1] целесообразность использования экспоненциально убывающей функции (3), расположенной достаточно близко к оси дискретного времени в последних итерациях. Следует отметить, что использование линейно убывающей функции сохраняет качество адаптации, но значительно увеличивает время расчетов. Пример построения адаптивной сетки для функции (3) приведен на рис. 3.

### Выводы

Представленная реализация нейросетевого алгоритма доказывает возможность построения адаптивных сеток на сложных физических областях, характеризующихся пористой структурой. В основе модифицированного алгоритма лежит идея чередования базового

алгоритма для граничных и внутренних узлов сетки в сочетании с упрощенной функцией соседства. Для получения лучшего результата в модифицированном алгоритме SOM рекомендуется в качестве параметра  $a$  Гауссовой функции расстояния использовать линейно убывающую функцию на первой макроитерации и экспоненциально убывающую в последующей. Вычислительные эксперименты по адаптации симметричной трехмерной полой области подтвердили рекомендации по выбору параметров обучения нейросети на односвязной плоской области [1].

### Список литературы

1. Архипов С.В. Модифицированные алгоритмы построения нейронной сети SOFM // Вестник БГУ. Математика и информатика. – 2011. – Вып. 9. – С. 61–68.
2. Годунов С.К., Проконов Г.П. О расчетах конформных отображений и построении разностных сеток // Журн. вычисл. математики и мат. физики. – 1967. – Т. 7. – С. 1031–1059.
3. Лисейкин В.Д., Лебедев А.С., Китаева И.А. Универсальный эллиптический метод построения разностных сеток. – Новосибирск: НГУ, 2004. – 266 с.
4. Нечаева О.И. Нейросетевой подход для построения адаптивных сеток // Нейроинформатика – 2006. – Ч. 2. – С. 172–179.
5. Нечаева О.И. Композиционный алгоритм для построения адаптивных сеток произвольной структуры // Сборник научных трудов Всероссийской научно-технической конференции Нейроинформатика-2007. – М.: МИФИ, 2007. – С. 72–79.
6. Хакимзянов Г.С., Шокин Ю.И., Барахнин В.Б., Шокина Н.Ю. Численное моделирование течений жидкости с поверхностными волнами. – Новосибирск: Изд-во СО РАН, 2001. – 394 с.
7. Gordon W.J., Thiel L.C. Transfinite mappings and their applications to grid generation. // Numerical Grid Generation, Appl. Mathematics and Computation. – 1982. – Vol. 2/3. – P. 171–192.

8. Koutnik J., Mazl R., Kulich M. Building of 3d environment models for mobile robotics using self-organization // In Proc. of The 9th International Conference on Parallel Problem Solving From Nature – PPSN-IX, Springer. – 2006. – P. 721–730.

9. Kohonen T. Self-organizing Maps // Springer Series in Information Sciences, Springer, Berlin, Heidelberg. – New York, 2001. – Vol. 30. – 501 p.

10. Kohonen T. K. Self-organization and associative memory. – New York: Springer Verlag, 1989. – 312 p.

11. Manevitz L., Yousef M. Finite-Element Mesh Generation Using Self-Organizing Neural Networks // Microcomputers in Civil Engineering. – 1997. – № 12. – P. 233–250.

12. Ritter H., Martinetz T., Schulten K. Neural Computation and Self-Organizing Maps: An Introduction. – New York: Addison-Wesley, 1992.

13. Thompson J.F., Warsi Z.U.A., Mastin C.W. Numerical grid generation, foundations and applications. – Amsterdam: North-Holland, 1985.

### References

1. Arhipov S.V. Modificirovannye algoritmy postroeniya nejronnoj seti SOFM // Vestnik BGU. Matematika i informatika. Vyp. 9. 2011. pp. 61–68.

2. Godunov S.K., Prokonov G.P. O raschetah konformnyh otobrazhenij i postroenii raznosnyh setok // Zhurn. vychisl. matematiki i mat. fiziki. T. 7. 1967. pp. 1031–1059.

3. Lisejkin V.D., Lebedev A.C., Kitaeva I.A. Universalnyj jellipticheskij metod postroeniya raznosnyh setok // Novosibirsk: NGU, 2004. 266 p.

4. Nechaeva O.I. Nejrosetevoj podhod dlja postroeniya adaptivnyh setok // Nejroin-formatika 2006. Ch. 2. 172–179 p.

5. Nechaeva O.I. Kompozicionnyj algoritim dlja postroeniya adaptivnyh setok proiz-volnoj struktury // Sbornik nauchnyh trudov Vserossijskoj nauchno-tehnicheskoy konferencii Nejroinformatika-2007. M.: MIFI, 2007. pp. 72–79.

6. Hakimzjanov G.S., Shokin Ju.I., Barahnin V.B., Shokina N.Ju. Chislennoe modelirovanie techenij zhidkosti s poverhnostnymi volnami. Novosibirsk: Izd-vo SO RAN, 2001, 394 p.

7. Gordon W.J., Thiel L.C. Transfinite mappings and their applications to grid generation // Numerical Grid Generation, Appl. Mathematics and Computation, Vol. 2/3. 1982. pp. 171–192.

8. Koutnik J., Mazl R., Kulich M. Building of 3d environment models for mobile robotics using self-organization // In Proc. of The 9th International Conference on Parallel Problem Solving From Nature PPSN-IX, Springer, 2006, pp. 721–730.

9. Kohonen T. Self-organizing Maps // Springer Series in Information Sciences, V.30, Springer, Berlin, Heidelberg, New York, 2001, 501 p.

10. Kohonen T.K. Self-organization and associative memory. New York: Springer Verlag. 1989. 312 p.

11. Manevitz L., Yousef M. Finite-Element Mesh Generation Using Self-Organizing Neural Networks // Microcomputers in Civil Engineering 12, 1997, pp. 233–250.

12. Ritter H., Martinetz T., Schulten K. Neural Computation and Self-Organizing Maps: An Introduction. New York: Addison-Wesley, 1992.

13. Thompson J.F., Warsi Z.U.A., Mastin C.W. Numerical grid generation, foundations and applications // Amsterdam: North-Holland. 1985.