

В. В. Пекунов, д-р техн. наук, инженер, e-mail: pekunov@mail.ru  
 ОАО "Информатика", г. Иваново

## Компактное описание вариативных полей физических величин в повторяющихся задачах моделирования атмосферного переноса загрязнителей

*Рассматривается задача сжатия полей данных, полученных при математическом моделировании воздушных потоков. На первом этапе поля кластеризуются по схожим значениям с применением алгоритма  $k$ -средних. На втором этапе каждый из полученных метакластеров в свою очередь кластеризуется по пространственным переменным. На третьем этапе каждый из полученных кластеров описывается набором нейронных сетей (в этом и состоит суть предлагаемой компрессии). Предлагается специальный алгоритм выбора оптимального набора нейронных сетей. Алгоритм использует технику "прогноз—коррекция". Прогноз реализуется путем экстраполяции с помощью специально построенной интерполирующей функции. Коррекция реализуется с использованием результатов приближения пробной нейронной сетью. Данная схема дает сжатие полей данных в 45...70 раз. Задача имеет существенную вычислительную сложность и параллельно решается на многоядерной системе.*

**Ключевые слова:** сжатие данных с потерями, кластеризация, нейронные сети, численное моделирование, адаптивный алгоритм, распараллеливание

### Введение

Рассмотрим задачу моделирования переноса загрязнителей в определенной сложной расчетной области. В практике расчета экологической безопасности такая задача может решаться неоднократно, с варьированием как направления и скорости ветра, так и расположения и интенсивности источников загрязнителей. При этом целесообразно прибегнуть к однократному решению повторяющейся подзадачи расчета картины воздушных потоков для каждого конкретного направления/скорости ветра с сохранением результатов расчета в файлах. Отметим, что в некоторых случаях указанная подзадача может решаться путем интерполяции по данным из сохраненных результатов расчета для иных, близких к текущему, направлений/скоростей ветра. При больших размерах расчетной сетки (от нескольких сотен до миллионов узлов для областей сложной формы, включающих, например, несколько улиц со строениями разной высоты) объем сохраняемых конечных данных (полученного стационарного распределения воздушных потоков — трех полевых компонент вектора скорости и поля турбулентной вязкости) даже для одной подзадачи при использовании вещественных значений одинарной точности (4-байтных) может достигать нескольких десятков мегабайт. С учетом того, что число подзадач определяется произведением числа возможных направлений ветра (15...30) на число градаций скорости ветра (10...15), его величина может составлять от 150 до 450 подзадач. В результате общий объем сохраняемых данных может достичь одного гигабайта и выше.

Поэтому актуальна задача генерации компактных описаний получаемых вариативных полей физических величин (компонент скорости и турбулентной вязкости). Как показывает практика, для этих данных максимальный коэффициент сжатия без потерь такими достаточно качественными алгоритмами, как RРМD и LZMA, не превышает 0,50...0,65. Для получения большей степени сжатия (например, с коэффициентом 0,01...0,02) необходимо воспользоваться прибли-

женными методами (сжатие с потерями), например, интерполяцией данных нейронными сетями с сохранением их весовых коэффициентов. Применение нейронных сетей также решит проблему получения интерполированных значений физических величин для случая таких направлений/скоростей ветра, для которых нет сохраненных (сжатых или несжатых) данных.

С вычислительной точки зрения задача сжатия может сводиться к кластеризации и обучению динамически определяемого набора нейронных сетей в каждом кластере. Это достаточно затратные процедуры, поэтому для их решения целесообразно использовать параллельные вычисления.

### Кластеризация

Поля физических величин обычно характеризуются наличием множества достаточно компактных групп схожих (принадлежащих некоему сравнительно небольшому интервалу) значений, взятых в узловых точках расчетной сетки. Это приводит к идее, что первый и второй этапы описания вариативного поля  $V(x, y, z, a)$ , где  $a$  — некий варьируемый параметр (например, направление ветра), должны состоять в двухслойной кластеризации: сначала метакластеры по значениям  $Q(x_i, y_j, z_k) = f(V(x_i, y_j, z_k, a))$  в узловых точках  $(i, j, k)$ , затем в пределах каждого из полученных метакластеров — кластеры по координатам. Можно попытаться описать поле в пределах каждого из полученных кластеров нейронной сетью прямого распространения. На входы сети подаются как значение параметра  $a$ , так и значения координат  $x_i, y_j, z_k$ , а выходом сети будет  $V(x_i, y_j, z_k, a)$  в узловой точке  $(i, j, k)$  при указанном значении  $a$ <sup>1</sup>.

<sup>1</sup> Заметим, что мы сознательно отказались от варианта, при котором сеть сразу же пытается предсказать все  $N_a$  значения: в таком случае вход для  $a$  отсутствует, но появляется  $N_a$  выходов, каждый  $p$ -й из которых выдает значение  $V(x, y, z, a_p)$ . Такой вариант, по нашему мнению, не обеспечивает должного качества интерполяции при небольшом числе нейронов и требует чрезвычайно большого времени на обучение.

Отметим, что поскольку первичная кластеризация должна выделять блоки с близкими значениями  $V(x, y, z, a)$  по всем возможным значениям  $a$ , то  $Q(x, y, z)$  должна являться некоей сверткой значений:

$$Q(x, y, z) = \sum_{p=1}^{N_a} w_p V(x, y, z, a_p), \text{ где } N_a \text{ — число воз-}$$

можных различных значений  $a$ ,  $w_p$  — относительная значимость  $p$ -го значения  $a$  в общем спектре значений этого параметра. Целесообразно, например, давать большие значения  $w_p$  для таких  $a_p$ , при которых вероятны повышенные ошибки интерполяции нейронной сетью. Такие значения могут быть назначены исходя из соображений исследователя или определены, например, путем простого эксперимента по попытке обучения  $N_a$  нейронных сетей, каждая  $p$ -я из которых ( $p = \overline{1, N_a}$ ) приближает все поле  $V(x, y, z, a_p)$  при фиксированном  $a_p$ , принимая на входы все узловые точки области [1]. По полученным средним погрешностям приближения  $\varepsilon_p$  исходных данных  $p$ -й сетью и выставляются веса:

$$w_p = w_{\min} + \frac{w_{\max} - w_{\min}}{\sum_{r=1}^{N_a} \varepsilon_r} \varepsilon_p$$

где  $w_{\min}$  и  $w_{\max}$  — соответственно минимальная и максимальная величины значимости.

### Построение нейросетевых описаний в кластерах

Третьим этапом описания вариативного поля  $V(x, y, z, a)$  в каждом кластере, как и указано выше, является его приближение нейронной сетью. При этом достаточно интересным является вопрос, следует ли воспользоваться одной более сложной сетью для всех значений  $a_p$  или же разбить множество  $\{a_p\}$  на подмножества, в каждом из которых применить более простую нейронную сеть. Как показали эксперименты по описанию полей скорости воздушных потоков, возможны оба варианта, причем второй требует меньше времени на обучение и дает лучшие результаты приблизительно в 1,5...2 раза чаще, что будет продемонстрировано ниже в следующем разделе. Однако в каждом конкретном случае необходим контекстный выбор разбиения  $\{a_p\}$ , что приводит к идее организации *адаптивного алгоритма, который выбирает вариант интерполяции* в пределах кластера. При этом, чтобы избежать полного перебора, целесообразно ограничиться лишь несколькими (3...4) основными вариантами, а для повышения скорости отбора вариантов воспользоваться предикцией погрешности для некоторых из них (по уже полученным данным), с ее частичной коррекцией по данным, полученным путем ограниченного расчета нескольких компонент новых вариантов.

Предлагается следующий адаптивный эвристический алгоритм.

1. Воспользуемся трехслойными нейронными сетями с экспоненциальными сигмодами в первых двух слоях и с линейной передаточной функцией в третьем выходном слое.

2. Определим начальное число нейронов при интерполяции единственной нейронной сетью. Для каждого значения  $a_p$ ,  $p = \overline{1, N_a}$ , найдем среднее по кластеру значение  $q_p$  поля  $V(x, y, z, a_p)$ , где  $(x, y, z) \in S$ , причем  $S$  — множество точек текущего рассматриваемого кластера. Получаем вектор значений  $q = (q_1, q_2, \dots, q_{N_a})$ , по которому можно определить набор точек  $(p, q_p)$ . По этому набору проводим интерполяцию полиномом достаточно высокого порядка  $r$ , где  $N_a/4 \leq r < N_a$ . Находим число экстремумов  $N_e$  полученного полинома (достаточно разбить область определения полинома на малые интервалы и подсчитать суммарное число интервалов, на краях которых производная полинома меняет знак или на одном из краев имеет нулевое значение).

Тогда начальное число нейронов первого слоя

$$N_1 := \lfloor \max(N_{1\min}, \min(N_{1\max}, \alpha N_e)) \rfloor,$$

где  $N_e$  — число экстремумов, а второго слоя

$$N_2 := \lfloor \max(N_{2\min}, \min(N_{2\max}, \beta N_e)) \rfloor,$$

где  $N_{1\min}$ ,  $N_{1\max}$  — минимально и максимально допустимые количества нейронов первого слоя,  $\alpha$  и  $\beta$  — эмпирические коэффициенты,  $N_{2\min}$ ,  $N_{2\max}$  — минимально и максимально допустимые количества нейронов второго слоя. В наших экспериментах использовались следующие значения:

$$N_{1\min} = 7, N_{2\min} = 3, N_{1\max} = 18, \\ N_{2\max} = 10, \alpha = 3,3, \beta = 1,4.$$

3. Задаем начальное число сетей  $A_1 := 1$ . Пусть  $r := 1$ .

4. Инициализируем массив ошибок  $\forall s: E_s := 0$ .

5. Если  $r > 2$ , то экстраполируем погрешность:

5.1. Пусть  $g(s) = \left( \sum_{m=0}^{r-1} a_m s^m \right)^{-1}$ . Задаем  $r - 1$  точек

вида  $(s; E_s)$ ,  $s = \overline{1, r-1}$ . Проводим по этим точкам интерполяцию функцией  $g(s)$  методом наименьших квадратов, в результате определяем коэффициенты  $a_0, \dots, a_{r-1}$ .

5.2. Определяем  $E_r := g(r)$ . Если  $E_r < 0$ , то  $E_r := 0$ .

5.3. Оцениваем производную  $D := (E_r - E_{r-1}) / (A_r - A_{r-1})$ .

5.4. Если  $D > 0$  или  $|D| < D_{\text{доп}}$  (здесь  $D_{\text{доп}}$  — минимально допустимое значение производной), то:

5.4.1. Предыдущее  $(r - 1)$ -е приближение с  $A_r/2$  сетями признается наиболее успешным.

5.4.2. Выход из алгоритма.

6. Равномерно распределяем значения  $a_p$ ,  $p = \overline{1, N_a}$ , по  $A_r$  нейронным сетям.

7. Цикл по  $q = \overline{1, A_r}$ :

7.1. Обучаем  $q$ -ю нейронную сеть по связанным с ней значениям  $(x, y, z, a_p)$  во всех узлах текущего кластера. Определяем среднюю погрешность  $E_{avr}$

7.2. Если  $q = 1$  и  $r > 2$ , то:

7.2.1. Корректируем предсказанную погрешность:  $\tilde{E} := 0,8E_{avr} + 0,2E_r$

7.2.2. Оцениваем скорректированное значение производной  $D := (\tilde{E} - E_{r-1}) / (A_r - A_{r-1})$ .

7.2.3. Если  $D > 0$  или  $|D| < D_{\text{доп}}$ , то:

7.2.3.1.  $E_r := E_{avr}$

7.2.3.2. Предыдущее ( $r - 1$ )-е приближение с  $A_r/2$  нейронными сетями является самым успешным.

7.2.3.3. Выход из алгоритма.

7.3.  $E_r := E_r + E_{avr}$

7.4. Конец цикла по  $q$ .

8. Определяем среднюю погрешность:  $E_r := E_r/A_r$ .

9. Если  $E_r \leq E_{\text{треб}}$  (здесь  $E_{\text{треб}}$  — требуемое минимально допустимое значение погрешности), то переходим к п. 15.

10.  $A_{r+1} := 2A_r$

11.  $r := r + 1$ .

12.  $N_1 := \max(N_{1\min}, \lfloor \gamma \cdot N_1 \rfloor)$ .

13.  $N_2 := \max(N_{2\min}, \lfloor \delta \cdot N_1 \rfloor)$ .

14. Если  $A_r \leq A_{\max}$ , то переходим к п. 5.

15. Алгоритм завершен, разбиение с  $A_r/2$  сетями является лучшим.

В наших экспериментах использовались следующие значения:

$$D_{\text{доп}} = \frac{0,05}{4}, E_{\text{треб}} = 0,075, \gamma = 0,6, \delta = 0,4, A_{\max} = 8.$$

### Экспериментальная часть

В качестве примера рассмотрим построение компактных кластерно-нейросетевых описаний полей скорости воздушных течений в трехмерной расчетной области сложной формы, представляющей участок, включающий улицу Göttinger Straße города Ганновер (Hannover, Германия, 52°20' северной широты, 9°40' восточной долготы) и несколько примыкающих улиц. На рис. 1 представлен соответствующий фрагмент карты Ганновера. Данные о конфигурации расчетной

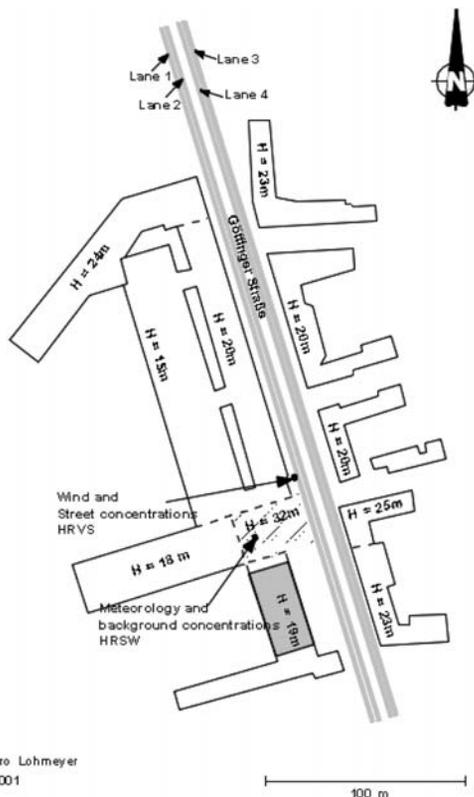


Рис. 1. Фрагмент карты города Ганновер, представляющий расчетную область

области и о различных условиях моделирования были получены из работ [2—4].

Имеющиеся данные (исходные для построения описаний) представляли результаты моделирования [5, 6] (получены на 128-ядерном кластере ИГЭУ на процессорах Opteron 2 ГГц, сеть передачи Gigabit Ethernet) с установившимися значениями полей скорости и турбулентной вязкости, полученные для 18 различных направлений ветра (от 20 до 360° с шагом 20°). Число узлов расчетной сетки составляло  $72 \times 86 \times 45 = 278\,640$ . Для хранения каждой из трех компонент полученных полей скорости использовались четырехбайтные значения одинарной точности. Соответственно, для хранения данных по каждой компоненте задействовались  $18 \times 72 \times 86 \times 45 \times 4 = 20\,062\,080$  байт, т. е. около 19 Мбайт.

Для построения компактных описаний была разработана параллельная программа (М-скрипт) для Octave 3.8.4 (Linux-версия), исполняющаяся на стандартной 16-ядерной машине платформы Google's Compute Engine.

На первом этапе использовался алгоритм  $k$ -средних (функция kmeans) для разбиения на 20 метакластеров. На втором этапе использовалось распараллеливание по метакластерам в соответствии с парадигмой "портфель задач" (с той же функцией pararrayfun): каждый метакластер на отдельном ядре тем же алгоритмом ( $k$ -средних) делился на 15 кластеров, таким образом, общее число кластеров достигло 300. На третьем этапе при обучении нейронных сетей методом обратного распространения ошибки также использовалось распараллеливание по принципу "портфель задач": один метакластер — одна задача (одно ядро). При этом задавалось максимальное число прогонов по обучающей выборке, равное 100. Кроме того, обучение прекращалось при достижении среднеквадратичной ошибки, равной 0,075.

Использовались все 16 ядер, время обработки каждой компоненты скорости занимало в среднем около 28 мин. Для сравнения: на одном ядре процессора Intel Atom 1,86 ГГц (Windows) время обработки одной компоненты занимало около 16,5 ч. С поправкой на различие тактовых частот процессоров (в Google's Compute Engine используются процессоры Xeon 2,6 ГГц, Linux) можно подсчитать реальные (объясняющиеся преимущественно эффектом распараллеливания) ускорение и эффективность распараллеливания. Они составляют соответственно  $\frac{16,5 \cdot 60 \cdot (1,86/2,6)}{28} = 25,3$  и  $\frac{16,5 \cdot 60 \cdot (1,86/2,6)}{28 \cdot 16} = 158\%$ .

Такой "суперлинейный" эффект объясняется, видимо, двумя факторами:

- а) большей эффективностью скомпилированного для Linux кода Octave по сравнению с кодом, полученным для Windows;
- б) более высокой производительностью процессора Intel Xeon в сравнении с Intel Atom.

В таблице представлены результаты построения компактных описаний по каждой из трех проекций вектора скорости.

Из таблицы очевидно, что при весьма высокой степени сжатия (в 43...71 раз) средняя абсолютная по-

### Результаты построения компактных описаний

Параметр	Проекция скорости на ось $x$	Проекция скорости на ось $y$	Проекция скорости на ось $z$
Число весов сетей	68 483	90 505	55 394
Число сетей	965	1371	522
Число нейронов	11 372	15 494	7833
Среднее число сетей на кластер	3,22	4,57	1,74
Объем памяти (байт), требуемый для хранения весовых коэффициентов, сжатой информации о распределении узлов по кластерам, о конфигурации сетей	351 306	454 010	283 002
Коэффициент сжатия	0,018	0,023	0,014
Число кластеров с одной сетью	27	15	138
Число кластеров с двумя сетями	159	92	136
Число кластеров с четырьмя сетями	73	93	24
Число кластеров с восемью сетями	41	100	2
Средняя абсолютная погрешность, м/с	0,218	0,217	0,081

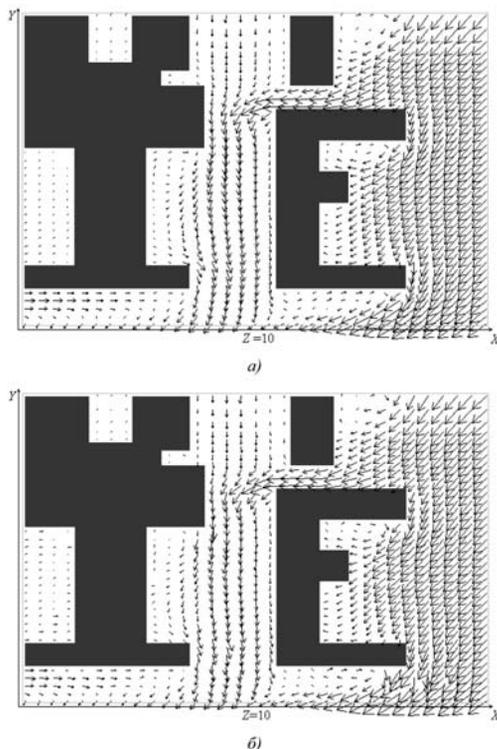


Рис. 2. Распределения воздушных потоков по результатам моделирования (а) и по описаниям (б)

грешность достаточно невелика с учетом того, что скорость воздушного течения в расчетной области принимала значения в диапазоне 0,1...9,5 м/с. Это подтверждается также визуально — на рис. 2 приведены картины скорости воздушных течений во фрагменте области на высоте 10 м, полученные в расчете (а) при направлении ветра в  $20^\circ$  и по компактным описаниям (б) расчетных данных. Очень показательны распределения чисел кластеров по числу задейство-

ванных сетей. По проекциям на оси  $Ox$  и  $Oy$  преобладают кластеры с более чем одной сетью, что объясняется достаточно сложным характером интерполируемых данных (подтверждается более высокой средней абсолютной погрешностью интерполяции). В то же время по проекции на ось  $Oz$ , напротив, доминируют кластеры с одной-двумя сетями, которые при сравнительно несложном характере исходных данных (погрешность интерполяции существенно ниже) уже дают требуемый результат.

На рис. 2, а, б отмечаются лишь небольшие искажения, что, вообще говоря, свидетельствует об адекватности построенных компактных кластерно-нейросетевых описаний. Соответственно, моделирование переноса загрязнителей при использовании восстановленных из описаний полей скорости также должно давать результаты, близкие к тем, которые можно получить при использовании исходных полей скорости.

### Выводы

В данной работе предложены основные принципы построения компактных кластерно-нейросетевых описаний, варьирующихся в зависимости от некоего параметра физических величин. Подробно описан адаптивный эвристический алгоритм построения описаний. Алгоритм отличается наличием процедуры предикции-коррекции погрешности для предполагаемого к использованию количества нейронных сетей.

На данных, полученных в результате моделирования воздушных течений на трехмерном участке, включающем несколько улиц города Ганновер, показано, что получаемые таким образом приближенные описания в 43...71 раз компактнее исходных данных, при этом погрешность восстановленных данных имеет вполне приемлемую величину. С качественной точки зрения восстановленные данные также весьма схожи с исходными данными. Такие данные вполне могут применяться для расчета атмосферного переноса загрязнителей в задачах, подобных описанным в [5, 6].

Показаны высокие степень ускорения и эффективность распараллеливания на стандартной 16-ядерной машине платформы Google's Compute Engine.

### Список литературы

1. Пекунов В. В. Локальные нейросетевые модели турбулентности // Мат. Междунар. науч.-техн. конф. "XVIII Бенардосовские чтения". Иваново, 2015. Т. 2. С. 331–334.
2. Ketzell M., Louka P., Sahn P., Guilloteau E., Sini J.-F. The use of computational fluid dynamics in modelling air quality in street canyons // Final report of the TMR research network TRAPOS "Optimisation of Modelling Methods for Traffic Pollution in Streets", submitted to the European Commission, July 2001. P. 86–93.
3. Ketzell M., Louka P., Sahn P., Guilloteau E., Sini J.-F., Moussiopoulos N. Intercomparison of Numerical Urban Dispersion Models — Part II: Street Canyon in Hannover, Germany // Proceedings of the 3rd Urban Air Quality conference — Loutraki 19–23 March 2000.
4. Schatzmann M., Leitl B., Liedtke J. Ausbreitung von Kfz-Abgasen in Straßenschluchten // Forschungsbericht FZKA-BWPLUS (Final report of the research project PEF 2 96 001). 1999. P. 1–132.
5. Пекунов В. В. Модель образования и распространения твердых, жидких и газообразных загрязнителей. Оптимальное распараллеливание // Математическое моделирование. 2009. Т. 21. № 3. С. 69–82.
6. Пекунов В. В. Новые методы параллельного моделирования распространения загрязнений в окрестности промышленных и муниципальных объектов // Дис. докт. тех. наук. Иваново, 2009. 274 с.

## The Compact Description of Variative Fields of Physical Variables in the Repetitive Tasks of Simulation of Atmospheric Pollution's Transfer

*This work deals with the problem of compactization of the data fields resulting from mathematical simulation of the atmospheric flows. In the first stage such fields are clustered (using k-means) by the similar values. In the second stage each such metacluster is clustered by space variables. In the third stage each resulting cluster is described by a sequence of neural networks (this is a proposed compression). A special algorithm of the selection of an optimal set of networks is proposed. This algorithm uses a prediction-correction technique. The prediction is realized by an extrapolation using a specially builded (interpolating) function. The correction is realized using a probe network results. The proposed scheme can give a compression of the data fields into 45–70 times. The task has a big calculating cost and is solved in parallel on the multicore machine.*

**Keywords:** lossy data compression, clustering, artificial neural networks, numerical simulation, adaptive algorithm, parallelization

### References

1. **Pekunov V. V.** Lokal'nye neyrosetevye modeli turbulentsnosti (Local neural network models of turbulence), *Mat. Mezhdunar. nauch.-tekh. konf. "XVIII Benardosovskie chteniya"* (Proceedings of the International Scientific-Technical Conference "XVIII Benardos readings"). Ivanovo, 2015, vol. 2, pp. 331–334 (in Russian).

2. **Ketzel M., Louka P., Sahm P., Guilloteau E., Sini J.-F.** The use of computational fluid dynamics in modelling air quality in street canyons, *Included in the final report of the TMR research network TRAPOS "Optimisation of Modelling Methods for Traffic Pollution in Streets"*, submitted to the European Commission, July 2001.

3. **Ketzel M., Louka P., Sahm P., Guilloteau E., Sini J.-F., Moussiopoulos N.** Intercomparison of Numerical Urban Dispersion Models. Part II: Street Canyon in Hannover, Germany, *The 3rd Urban Air Quality conference — Loutraki 19–23 March 2000*.

4. **Schatzmann M., Leitl B., Liedtke J.** *Ausbreitung von Kfz-Abgasen in Straßenschluchten / Forschungsbericht FZKA-BWPLUS* (Final report of the research project PEF 2 96 001). 1999.

5. **Pekunov V. V.** Model' obrazovaniya i rasprostraneniya tverdykh, zhidkikh i gazoobraznykh zagryazniteley. Optimal'noe rasparrallelvaniye (Model of creation and propagation of solid, liquid and gaseous pollutants. Optimal Parallelization), *Matematicheskoe modelirovaniye* [Mathematical Modelling], 2009, vol. 21, no. 3, pp. 69–82 (in Russian).

6. **Pekunov V. V.** Novye metody parallel'nogo modelirovaniya rasprostraneniya zagryazneniy v okrestnosti promyshlennykh i munitsipal'nykh ob"ektov (New methods of parallel modeling of pollution propagation near the plants and city objects), *Dis. dokt. tekhn. nauk* [Thesis of doctor of engineering]. Ivanovo, 2009 (in Russian).

УДК 004.5; 004.94

**С. Д. Кургалин**, д-р физ.-мат. наук., профессор, kurgalin@bk.ru,

**Я. А. Туровский**, канд. мед. наук, доц., yaroslav\_turovsk@mail.ru,

**С. В. Борзунов**, канд. физ.-мат. наук, доц., borzunov@cs.vsu.ru

Воронежский государственный университет, Воронеж

**А. А. Адаменко**, аспирант, adamenko.artem@gmail.com

Воронежский государственный университет инженерных технологий, Воронеж

## Теоретические аспекты оптимизации эволюционного обучения нейрочипов с использованием "изолятов"\*

*Для эволюционного подхода к обучению нейрочипов, применяемых для восстановления поврежденных нервной ткани, разработан алгоритм оптимизации, в основе которого лежит метод биологической изоляции в популяции. Предложено использовать понятие "изолята" — множества дочерних искусственных нейронных сетей (ИНС) одного поколения, имеющих одну и ту же родительскую ИНС. Представлены теоретические обоснования применения "изолятов" на основе модели линейного города. Они могут быть использованы в большинстве алгоритмов обучения ИНС без учителя, основывающихся на эволюции ИНС путем скрещивания их весовых коэффициентов. Разработанный алгоритм может быть применен при конструировании нейрочипов, используемых для восстановления поврежденной нервной ткани. Он позволяет в большинстве случаев определить правильное направление изменения весовых коэффициентов связей ИНС для уменьшения числа ошибок классификации и достигнуть лучших показателей обучения нейрочипа по сравнению с применением эволюционного алгоритма без "изолятов", что в конечном итоге приводит к улучшению результата восстановления поврежденной нервной ткани.*

**Ключевые слова:** нейрочип, нервная ткань, искусственная нейронная сеть, эволюционный алгоритм, моделирование обучения

\* Работа выполнена при частичной финансовой поддержке РФФИ (грант 16-29-08342-офи\_м).