

УДК 004.855.5

## РАЗРАБОТКА И ИССЛЕДОВАНИЕ ГИБРИДНЫХ ВЕРСИЙ АЛГОРИТМА РОЯ ЧАСТИЦ НА ОСНОВЕ АЛГОРИТМОВ ПОИСКА ПО СЕТКЕ

Л. А. Демидова, д.т.н., профессор кафедры ВПМ РГРТУ; demidova.liliya@gmail.com.

И. А. Клюева, аспирант РГРТУ; i.aleschenko@yandex.ru.

*Рассматривается подход к решению задачи безусловной оптимизации, основанный на гибридизации алгоритма роя частиц (Particle Swarm Optimization Algorithm, PSO-algorithm) и алгоритма поиска по сетке. Целью работы является разработка гибридных версий PSO-алгоритма и исследование их поисковых характеристик. В работе представлены две гибридные версии базового PSO-алгоритма, предполагающие использование соответственно классического алгоритма Grid Search (GS) и алгоритма Design of Experiment (DOE). В качестве базового предлагается использовать канонический PSO-алгоритм. Приведены результаты экспериментальных исследований, подтверждающие эффективность применения предложенных гибридных версий базового PSO-алгоритма при решении оптимизационных задач. При этом выполнен сравнительный анализ основных показателей качества базового PSO-алгоритма и его гибридных версий при решении задачи поиска глобального оптимума ряда тестовых функций. Кроме того, показана целесообразность применения гибридных версий базового PSO-алгоритма с целью сокращения временных затрат на поиск оптимальных значений параметров SVM-классификатора.*

**Ключевые слова:** алгоритм роя частиц, алгоритм поиска по сетке, гибридный алгоритм, тестовая функция, классификация, SVM-классификатор, параметры оптимизации, радиальная базисная функция ядра.

DOI: 10.21667/1995-4565-2016-57-3-105-116

### Введение

В настоящее время широкое применение находят алгоритмы оптимизации, созданные по образу существующих в природе биологических систем. Известны различные наименования таких алгоритмов: эволюционные, поведенческие, роевые, интеллектуальные, метаэвристические, биоинспирированные. Основная идея этих алгоритмов оптимизации заключается в практическом использовании каких-либо принципов биологических систем, эволюционированных природой в течение длительного времени.

Большинство эволюционных алгоритмов оперирует множествами простых существей, моделируя интеллектуальное поведение популяции, в которой каждая особь представляет некоторое альтернативное приближенное решение. Наиболее известными эволюционными алгоритмами являются генетический алгоритм, алгоритм роя частиц, муравьиный алгоритм, пчелиный алгоритм [1].

На протяжении долгого периода времени при решении трудоемких задач оптимизации наиболее востребованными являлись генетические алгоритмы [2]. Однако в последние годы

всё большее применение при решении различных прикладных задач оптимизации находит алгоритм роя частиц (Particle Swarm Optimization (PSO) algorithm) [1-3,5-10,14,15], предложенный в 1995 году Дж. Кеннеди и Д.К. Эберхартом в качестве алгоритма оптимизации непрерывных нелинейных функций.

PSO-алгоритм, несмотря на простоту реализации, и, вследствие этого, низкую алгоритмическую сложность, по эффективности может конкурировать со многими эволюционными алгоритмами оптимизации.

Однако следует отметить, что эффективность работы PSO-алгоритма в каждом конкретном случае зависит от правильности настройки его параметров.

В настоящее время известны различные способы повышения эффективности базового PSO-алгоритма, которые можно разделить на метаоптимизационные и комбинационные. Метаоптимизационные способы предполагают настройку и управление свободными параметрами PSO-алгоритма. Комбинационные способы предлагают различные варианты объединения PSO-алгоритма с другими алгоритмами оптимизации

с целью компенсации недостатков первого за счёт достоинств вторых [8].

В данной работе предлагается реализовать комбинационный способ повышения эффективности базового PSO-алгоритма посредством разработки его гибридных версий с применением того или иного алгоритма поиска по сетке. В частности, предполагается использовать два алгоритма поиска по сетке: классический алгоритм Grid Search (GS-алгоритм) [11] и алгоритм Design of Experiment (DOE-алгоритм) [11].

При этом будет выполнено исследование разработанных гибридных версий базового PSO-алгоритма. Планируется выполнить оценку значений основных показателей качества базового PSO-алгоритма и его гибридных версий при решении задачи поиска глобального оптимума ряда тестовых функций и провести их сравнительный анализ. Кроме того, аналогичные исследования будут проведены при решении задачи подбора оптимальных значений параметров SVM-классификатора.

### Теоретическая часть

В основу PSO-алгоритма положен тот факт, что частицы при формировании роя стремятся к некоторому центру «притяжения» (глобально оптимальному решению) [1-3,5-10,14,15].

В PSO-алгоритме пространство поиска заполняется популяцией частиц, каждая из которых в конкретный момент времени имеет в пространстве параметров решаемой задачи некоторое положение и скорость. Для каждого положения частицы роя вычисляется соответствующее значение целевой функции, на основе которого по определенным правилам (согласно расчетным формулам), вычисляются новые координаты (новое положение) и новая скорость частицы в пространстве поиска.

Каждая частица имеет память: частица запоминает свое лучшее положение в рое (лучшие координаты с точки зрения расстояния до глобального оптимума, полученные самой частицей). Кроме того, частицы способны обмениваться информацией: информация о глобально «лучшей» позиции среди всех частиц (лучшие координаты с точки зрения расстояния до глобального оптимума, полученные всеми частицами в совокупности) передается другим частицам.

С учетом информации, хранящейся в памяти частицы, на каждой итерации рассчитывается ее новая скорость, в зависимости от которой частица изменяет свое положение в пространстве поиска.

Пусть  $n$  – размерность пространства поиска (то есть  $n$  – количество параметров оптимизации), а  $m$  – количество частиц в рое. Координаты  $i$ -й частицы ( $i = \overline{1, m}$ ) в  $n$ -мерном пространстве поиска задаются вектором  $\vec{x}_i = (x_i^1, x_i^2, \dots, x_i^n)$ , который определяет некоторый набор значений параметров оптимизации. Значения координат (положение в пространстве поиска)  $i$ -й частицы ( $i = \overline{1, m}$ ) на каждой итерации PSO-алгоритма изменяются под влиянием ее вектора скорости  $\vec{v}_i \in R^n$ :  $\vec{v}_i = (v_i^1, v_i^2, \dots, v_i^n)$ .

В классической версии PSO-алгоритма коррекция вектора скорости  $i$ -й частицы ( $i = \overline{1, m}$ ) вычисляется в соответствии с формулой [2]:

$$\vec{v}_i(t+1) = \vec{v}_i(t) + \varphi_p r_p (\vec{p}_i(t) - \vec{x}_i(t)) + \varphi_g r_g (\vec{g}(t) - \vec{x}_i(t)), \quad (1)$$

где  $\vec{v}_i(t)$  – вектор скорости  $i$ -й частицы ( $i = \overline{1, m}$ ) на  $t$ -й итерации алгоритма;  $\vec{x}_i(t)$  – вектор координат  $i$ -й частицы ( $i = \overline{1, m}$ ), задающий её положение в рое на  $t$ -й итерации алгоритма;  $\vec{p}_i(t)$  – вектор координат лучшего положения, найденного  $i$ -й частицей за все время ее существования;  $\vec{g}(t)$  – вектор координат лучшего положения, найденного всеми частицами (глобально «лучшее» положение среди всех частиц в рое);  $r_p, r_g$  – случайные числа в интервале  $(0, 1)$ , которые вносят элемент случайности в процессе поиска;  $\varphi_p, \varphi_g$  – личный и глобальный коэффициенты ускорения частиц, которые являются константами и масштабируют случайные числа  $r_p$  и  $r_g$  (при этом глобальный коэффициент ускорения  $\varphi_g$  управляет воздействием глобальной лучшей позиции на скорости всех частиц, а личный коэффициент ускорения  $\varphi_p$  – воздействием личной лучшей позиции на скорость некоторой частицы) [2].

После коррекции скорости частицы выполняется коррекция ее текущего положения (координат в пространстве поиска):

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \vec{v}_i(t+1). \quad (2)$$

Затем для каждой  $i$ -й частицы ( $i = \overline{1, m}$ ) рассчитывается новое значение целевой функции и определяется «лучшее» положение этой частицы в рое. В задаче поиска минимума функции вида  $f(x) \rightarrow \min_{x \in R^n}$  под «лучшим» положением частицы будем понимать точку пространства поиска, в которой на текущей итерации PSO-алгоритма достигается минимальное значение целевой функции.

Если в результате проверки оказалось, что

новое положение  $i$ -й частицы стало лучшим среди всех точек пространства поиска, где она побывала, на  $t+1$  итерации PSO-алгоритма производится обновление вектора  $\vec{p}_i(t)$ . Затем, среди всех новых точек пространства поиска осуществляется проверка на нахождение нового глобально «лучшего» положения в рое: в случае его обнаружения обновляется вектор  $\vec{g}(t)$  и запоминается соответствующее значение целевой функции в данной точке пространства поиска.

В настоящее время известны различные версии PSO-алгоритма. Традиционное применение получила одна из самых распространенных версий – каноническая, в которой предлагается выполнять нормировку коэффициентов ускорения  $\varphi_p$  и  $\varphi_g$ , чтобы сходимость алгоритма не так сильно зависела от выбора их значений [2]. При этом коррекция вектора скорости  $\vec{v}_i(t)$   $i$ -й частицы ( $i = \overline{1, m}$ ) на  $t$ -й итерации PSO-алгоритма выполняется в соответствии с формулой:

$$\vec{v}_i(t+1) = \eta \cdot [\vec{v}_i(t) + \varphi_p r_p (\vec{p}_i(t) - \vec{x}_i(t)) + \varphi_g r_g (\vec{g}(t) - \vec{x}_i(t))], \quad (3)$$

где  $\eta = 2 \cdot K / |2 - \varphi - \sqrt{\varphi^2 - 4 \cdot \varphi}|$ ;  $\varphi = \varphi_p + \varphi_g$  ( $\varphi > 4$ );  $\eta$  – коэффициент сжатия;  $K$  – некоторый масштабирующий коэффициент, принимающий значения из интервала (0, 1).

При использовании формулы (3) для коррекции вектора скорости гарантируется сходимость PSO-алгоритма и нет необходимости в явном контроле скорости частиц [2].

Можно выделить несколько подходов к повышению эффективности классического PSO-алгоритма: гибридизация алгоритма с другими алгоритмами оптимизации, уменьшение вероятности преждевременной сходимости посредством изменения характеристик движения частиц, а также динамическое изменение параметров алгоритма во время оптимизации [8].

В последние годы всё большее применение находят подходы, реализующие гибридизацию PSO алгоритма с другими алгоритмами оптимизации.

В данной работе предлагается реализовать две гибридные версии PSO-алгоритма с применением двух алгоритмов поиска по сетке: классического GS-алгоритма [11] и DOE-алгоритма [11].

В GS-алгоритме на диапазонах изменения параметров оптимизации с некоторым шагом по каждому параметру строится сетка (рисунок 1, частный случай в пространстве поиска D-2) и

выполняется оценка эффективности всех комбинаций значений параметров оптимизации на сетке. В результате исследуются все узлы сетки. Преимущество поиска по узлам сетки заключается в скрупулезности нахождения глобально оптимального решения.

Альтернативным алгоритмом поиска по сетке является DOE-алгоритм. В этом алгоритме поиск глобально оптимального решения осуществляется не во всех узлах сетки, а в определенным образом выбранных узлах (рисунок 2, частный случай в пространстве поиска D-2).

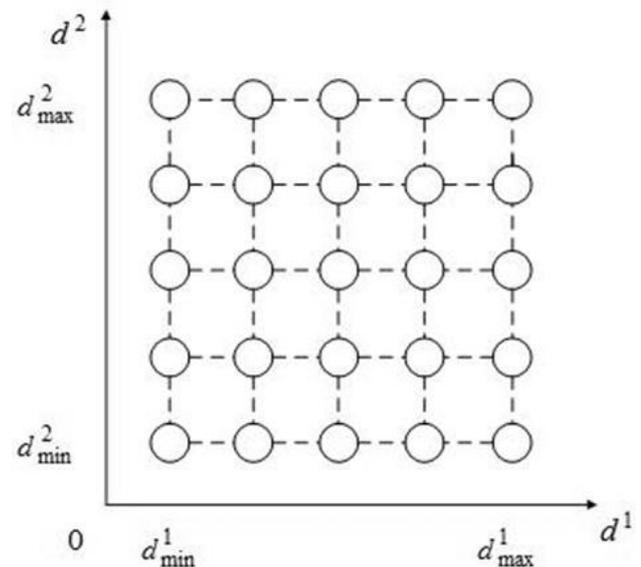


Рисунок 1 – Формирование сетки в GS-алгоритме

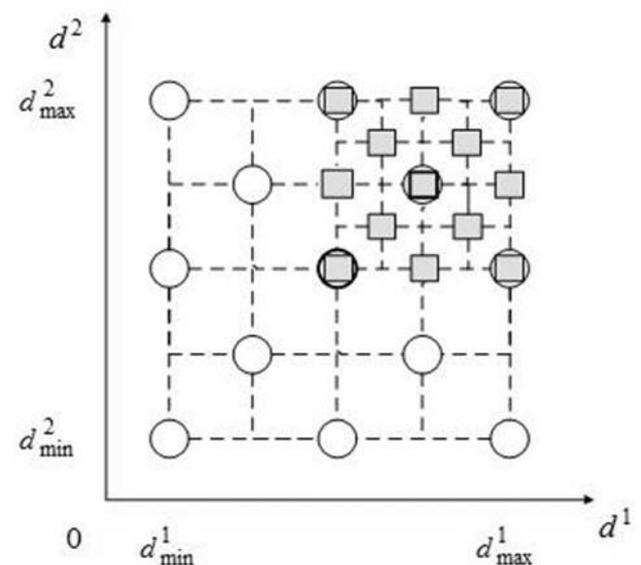


Рисунок 2 – Формирование сетки в DOE-алгоритме

Данный алгоритм является итерационным. На каждой итерации алгоритма вокруг некоторого центрального узла строится сетка (например, сетка из 13 узлов в пространстве поиска D-2), затем выбирается лучший из этих узлов (в соот-

ветствии с лучшим значением оптимизируемой [целевой] функции), и вокруг этого узла как центра сетки строится новая сетка (например, сетка из 13 узлов в пространстве поиска D-2). При этом при переходе к новой итерации размеры пространства поиска по сетке уменьшаются вдвое и сетка сужается. Подробное описание DOE-алгоритма приведено в изложении соответствующей гибридной версии PSO-алгоритма.

Достоинства DOE-алгоритма заключаются в следующем.

1. Границы поиска многократно совершенствуются, пока условия остановки поиска не удовлетворены. После каждой итерации DOE-алгоритма область поиска сужается и уточняется так, чтобы ее центром был «лучший» найденный узел, которому соответствует лучшее значение целевой функции.

2. Если процесс поиска выходит за пределы изначально заданных (допустимых) границ диапазонов поиска, то новые границы диапазонов поиска будут определены таким образом, чтобы новое пространство поиска в DOE-алгоритме содержалось в пределах допустимых границ диапазонов поиска.

При создании гибридной версии PSO-алгоритма предлагается на каждой итерации PSO-алгоритма выполнять уточнение положения (то есть координат) глобально «лучшей» частицы в рое с использованием того или иного алгоритма поиска по сетке с последующим обновлением текущей популяции частиц роя. При этом из роя должна быть удалена «худшая» частица (то есть частица с «худшим» значением оптимизируемой [целевой] функции), и вместо нее должна быть добавлена «лучшая» частица, найденная алгоритмом поиска по сетке.

В результате в гибридной версии PSO-алгоритма ускорение поиска глобально-оптимального решения может быть достигнуто благодаря:

– дополнительному поиску по сетке в области потенциального глобально «лучшего» положения частиц в рое;

– обновлению популяции роя частиц и удалению «худших» частиц.

Гибридная версия PSO-алгоритма может быть реализована в соответствии со следующей последовательностью шагов.

Шаг 1. Определить начальные характеристики частиц в рое (то есть координаты и скорости частиц); инициализировать значения настраиваемых параметров PSO-алгоритма: в случае использования канонического PSO-алгоритма параметрами являются число частиц в рое, количество итераций PSO-алгоритма, границы

диапазонов поиска, коэффициенты личного и глобального ускорения, масштабирующий коэффициент.

Шаг 2. Реализовать один шаг PSO-алгоритма. Скорректировать для каждой  $i$ -й частицы ( $i = \overline{1, m}$ ) скорость  $\vec{v}_i(t)$  и текущее положение (координаты)  $\vec{x}_i(t)$ . Найти координаты глобально «лучшей» частицы в рое (с лучшим значением целевой функции), определяющие вектор  $\vec{g}(t)$ .

Шаг 3. Определить границы диапазонов поиска для одного из алгоритмов поиска по сетке (GS-алгоритма или DOE-алгоритма). При этом определить диапазоны  $[d_{\min}^j, d_{\max}^j]$  ( $j = \overline{1, n}$ ) с учетом диапазонов  $[r_{\min}^j, r_{\max}^j]$  разброса частиц в рое на текущей итерации PSO-алгоритма. В качестве значений координат  $\chi^j$  ( $j = \overline{1, n}$ ) «главного» (центрального) узла сетки использовать значения координат глобально «лучшей» частицы в рое, хранящиеся в векторе  $\vec{g}(t)$ . Такой узел рассматривается в качестве центроида, вокруг которого строится сетка. Минимальное расстояние от лучшей частицы роя (центроида сетки) до границ диапазонов разброса находится как:

$$l^j = \min\{\chi^j - r_{\min}^j, r_{\max}^j - \chi^j\}, \quad (4)$$

а границы диапазонов поиска по сетке определяются как:

$$d_{\min}^j = \chi^j - l^j, \quad (5)$$

$$d_{\max}^j = \chi^j + l^j. \quad (6)$$

Шаг 4. Уточнить координаты глобально «лучшей» частицы в рое посредством выполнения одного из алгоритмов поиска по сетке (GS-алгоритма или DOE-алгоритма). Проверить, достигнуто ли реально уточнение координат глобально «лучшей» частицы роя. Если уточнение достигнуто (новое решение получено), то перейти к шагу 5, иначе – перейти к шагу 6.

Шаг 5. Переопределить вектор  $\vec{g}(t)$ , приняв в качестве нового глобально-оптимального решения на текущей итерации PSO-алгоритма решение, полученное при реализации алгоритма поиска по сетке на шаге 4. Выполнить обновление популяции частиц в рое: удалить «худшую» частицу роя и вместо нее добавить «лучшую» частицу, найденную на шаге 4.

Шаг 6. Выполнить переход к шагу 7, если выполнены условия останова PSO-алгоритма (достигнуто максимальное количество итераций PSO-алгоритма или найден глобальный оптимум с заданной точностью), иначе – перейти к шагу 2.

Шаг 7. Принять значения координат «луч-

шей» частицы роя в качестве искомого глобально-оптимального решения и завершить работу гибридной версии PSO-алгоритма.

Ниже рассмотрены особенности реализации алгоритмов поиска по сетке, использующихся на шаге 4 предложенной гибридной версии PSO-алгоритма.

В GS-алгоритме диапазоны поиска  $[d_{\min}^j, d_{\max}^j]$  ( $j = \overline{1, n}$ ), найденные на основе формул (5) и (6) на шаге 3 гибридной версии PSO-алгоритма, разбиваются на заданное количество интервалов, в результате чего определяются узлы сетки. Затем в каждом узле сетки вычисляется значение оптимизируемой (целевой) функции. В результате реализации GS-алгоритма будет определен «лучший» узел с «лучшим» значением целевой функции. Координаты этого узла в дальнейшем могут быть использованы в качестве координат новой глобально «лучшей» частицы в рое.

DOE-алгоритм обычно используется для решения задач оптимизации в пространстве поиска D-2, однако легко может быть адаптирован для выполнения расчетов в пространстве произвольной размерности  $n$ . Поскольку в дальнейшем планируется использовать гибридную версию PSO-алгоритма с применением DOE-алгоритма именно для решения задач оптимизации в пространстве поиска D-2 (то есть при  $n = 2$ ), а также по причине хорошей наглядности реализации DOE-алгоритма в этом пространстве, дальнейшее описание реализации DOE-алгоритма на шаге 4 гибридной версии PSO-алгоритма приведено для частного случая в пространстве поиска D-2.

Шаг 1. Определить в границах диапазонов  $[d_{\min}^j, d_{\max}^j]$  ( $j = 1, 2$ ) 13 узлов сетки (на рисунке 2 узлы первой итерации DOE-алгоритма помечены маркерами круглой формы белого цвета, а узлы второй итерации – маркерами квадратной формы серого цвета, при этом узлы, участвующие на нескольких итерациях, помечены дважды маркерами круглой и квадратной формы). При этом центральный узел (центроид сетки) с координатами  $\chi^j$  ( $j = 1, 2$ ) (пример на рисунке 2 - маркер круглой формы с выделенным контуром) соответствует глобально «лучшей» частице роя, а ширина диапазонов поиска на текущей итерации DOE-алгоритма определяется как  $S^j = d_{\max}^j - d_{\min}^j$  ( $j = 1, 2$ ).

Координаты узлов такой сетки определяются следующим образом (при движении по сетке из нижнего левого узла снизу вверх слева направо):

$$\begin{aligned} & [\chi^1 - S^1/2, \chi^2 - S^2/2], [\chi^1 - S^1/2, \chi^2 + S^2/2], \\ & [\chi^1 + S^1/2, \chi^2 + S^2/2], [\chi^1 + S^1/2, \chi^2 - S^2/2], \\ & [\chi^1 - S^1/2, \chi^2], [\chi^1, \chi^2 + S^2/2], \\ & [\chi^1 + S^1/2, \chi^2], [\chi^1, \chi^2 - S^2/2], \\ & [\chi^1 - S^1/4, \chi^2 - S^2/4], [\chi^1 - S^1/4, \chi^2 + S^2/4], \\ & [\chi^1 + S^1/4, \chi^2 + S^2/4], [\chi^1 + S^1/4, \chi^2 - S^2/4], \\ & [\chi^1, \chi^2]. \end{aligned}$$

Шаг 2. Вычислить значение целевой функции в каждом узле сетки и найти координаты  $\phi^j$  ( $j = 1, 2$ ) узла с «лучшим» значением целевой функции.

Шаг 3. Переопределить ширину диапазонов поиска как  $S^j/2$  ( $j = 1, 2$ ) и использовать вычисленные таким образом значения в качестве новых значений  $S^j$  ( $j = 1, 2$ ) для следующей итерации DOE-алгоритма.

При этом новые границы диапазонов поиска по сетке для следующего шага переопределяются как:

$$d_{\min}^j = \phi^j - S^j/2, \quad (7)$$

$$d_{\max}^j = \phi^j + S^j/2. \quad (8)$$

Шаг 4. Перейти к шагу 1, если не исчерпано количество итераций DOE-алгоритма, иначе – завершить работу алгоритма. При этом в качестве новых координат центрального узла сетки  $\chi^j$  ( $j = 1, 2$ ) (пример на рисунке 2- маркер квадратной формы с выделенным контуром) принимаются значения координат «лучшего» узла  $\phi^j$  ( $j = 1, 2$ ), найденного на текущей итерации DOE-алгоритма.

Следует отметить, что границы диапазонов  $[d_{\min}^j, d_{\max}^j]$  ( $j = 1, 2$ ) для первой итерации DOE-алгоритма вычисляются на основе формул (5) и (6) на шаге 3 гибридной версии PSO-алгоритма, а для всех остальных итераций DOE-алгоритма – на основе формул (7) и (8) на шаге 3 самого DOE-алгоритма.

При реализации DOE-алгоритма выполняется контроль за допустимостью вновь вычисленных границ диапазонов поиска. Если на некоторой текущей итерации DOE-алгоритма координаты «лучшего» найденного узла оказались вблизи текущих границ диапазонов поиска по сетке, то при построении сетки на следующей итерации DOE-алгоритма возможен выход за пределы изначально заданных (допустимых) в гибридной версии PSO-алгоритма границ диапа-

зонов поиска  $[range_{\min}^j, range_{\max}^j]$  ( $j=1,2$ ). В случае если после вычисления по формулам (7) и (8) новых границ диапазонов поиска по сетке  $[d_{\min}^j, d_{\max}^j]$  ( $j=1,2$ ) оказалось, что выполняется одно из условий  $d_{\min}^j < range_{\min}^j$  при некотором  $j = j^* \in \{1,2\}$  или  $d_{\max}^j > range_{\max}^j$  при некотором  $j = j^* \in \{1,2\}$ , т.е. произошел выход за пределы изначально заданных (допустимых) в гибридной версии PSO-алгоритма границ диапазонов поиска, сетка сужается до новых границ диапазонов поиска по формулам:

– если  $d_{\min}^j < range_{\min}^j$  при некотором  $j = j^* \in \{1,2\}$ , то

$$d_{\min}^{j^*} = \phi^{j^*} - (\phi^{j^*} - range_{\min}^{j^*}), \quad (9)$$

$$d_{\max}^{j^*} = \phi^{j^*} + (\phi^{j^*} - range_{\min}^{j^*}); \quad (10)$$

– если  $d_{\max}^j > range_{\max}^j$  при некотором  $j = j^* \in \{1,2\}$ , то

$$d_{\min}^{j^*} = \phi^{j^*} - (range_{\max}^{j^*} - \phi^{j^*}), \quad (11)$$

$$d_{\max}^{j^*} = \phi^{j^*} + (range_{\max}^{j^*} - \phi^{j^*}). \quad (12)$$

В настоящее время для решения широкого спектра классификационных задач в различных прикладных областях успешно применяется SVM-алгоритм (Support Vector Machines, SVM) [2-9,11-13], являющийся одним из наиболее популярных алгоритмов машинного обучения по прецедентам («обучения с учителем»).

SVM-алгоритм предполагает выполнение обучения, тестирования и классификации. При удовлетворительном качестве обучения и тестирования полученный SVM-классификатор может быть применен для классификации новых объектов [3].

Правильный выбор значений параметров SVM-классификатора позволяет повысить качество принимаемых классификационных решений. В простейшем случае поиск искомых значений параметров SVM-классификатора может быть осуществлен в процессе перебора возможных комбинаций значений параметров SVM-классификатора, что требует значительных вычислительных затрат.

В настоящее время известны различные подходы к решению задачи подбора оптимальных значений параметров SVM-классификатора.

В последние годы всё большее применение находят поисковые алгоритмы стохастической оптимизации, в том числе PSO-алгоритм и его модификации [1-3,5-10,14-15]. В связи с этим предлагаемые гибридные версии PSO-алгоритма также могут быть применены для поиска опти-

мальных значений параметров SVM-классификатора.

При разработке SVM-классификатора в случае нелинейной разделимости классов при использовании радиальной базисной функции ядра [2-7,9,11-12] предлагается применять гибридные версии PSO-алгоритма для поиска оптимальных значений двух параметров SVM-классификатора: параметра регуляризации  $C$  и коэффициента функции ядра  $\sigma$ . При этом радиальная базисная функция ядра определяется как:

$$k(x_g, x_\tau) = \exp(-\langle x_g - x_\tau, x_g - x_\tau \rangle / (2 \cdot \sigma^2)), \quad (13)$$

где  $\langle x_g, x_\tau \rangle$  – скалярное произведение векторов  $x_g$  и  $x_\tau$ ;  $\sigma$  – коэффициент радиальной базисной функции ядра ( $\sigma > 0$ ).

Радиальная базисная функция ядра  $k(x_g, x_\tau)$  позволяет оценить близости объекта  $x_g$  к объекту  $x_\tau$ .

Оценка качества SVM-классификатора может быть выполнена с применением различных показателей качества: показатель, учитывающий данные кросс-проверки; показатели точности и полноты классификации; показатель, основанный на анализе ROC-кривой; показатель Карра и т.п. [7].

### Экспериментальные исследования

При выполнении экспериментальных исследований были использованы:

- канонический PSO-алгоритм (далее базовый PSO-алгоритм);
- гибридная версия базового PSO-алгоритма на основе классического GS-алгоритма поиска по сетке (далее PSO-GS-алгоритм);
- гибридная версия базового PSO-алгоритма на основе DOE-алгоритма поиска по сетке (далее PSO-DOE-алгоритм).

Программная реализация данных алгоритмов была выполнена с использованием языка программирования Python (версия 2.7).

Сравнительный анализ указанных выше трёх оптимизационных алгоритмов был выполнен с применением тестовых функций, информация о которых приведена в таблице 1. При этом при выполнении расчетов размерность пространства поиска полагалась равной 2 ( $n = 2$ ).

В ходе исследований для каждой тестовой функции было выполнено по 100 случайных реализаций (запусков) каждого оптимизационного алгоритма. При этом поиск глобального оптимума целевой функции в течение одного запуска алгоритма осуществлялся до момента достижения максимального числа итераций (количества поколений), равного 1000, либо нахождения пер-

вого значения целевой функции с заданной точностью поиска глобального оптимума, равной 0.001 ( $\varepsilon = 0.001$ ).

Оценка эффективности оптимизационных алгоритмов осуществлялась с использованием следующих показателей качества.

Таблица 1 – Тестовые функции

Функция	Формула, размерность пространства поиска, диапазоны поиска, глобальный оптимум
<b>Растринга</b> (рисунок 3, а)	$f_1(x) = \sum_{i=1}^n [10 + x_i^2 - 10 \cos(2\pi x_i)]$ ; $n = 2$ ; $x_1 \in [-5.12, 5.12], x_2 \in [-5.12, 5.12]$ ; глобальный минимум $f_1^{\min} = 0$ достигается в точке (0, 0) (то есть при $x_1 = 0$ и $x_2 = 0$ ).
<b>Розенброка</b> (рисунок 3, б)	$f_2(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$ ; $n = 2$ ; $x_1 \in [-30, 30], x_2 \in [-30, 30]$ ; глобальный минимум $f_2^{\min} = 0$ достигается в точке (1, 1) (то есть при $x_1 = 1$ и $x_2 = 1$ ).
<b>Сферы</b> (рисунок 3, в)	$f_3(x) = \sum_{i=1}^n x_i^2$ ; $n = 2$ ; $x_1 \in [-100, 100], x_2 \in [-100, 100]$ ; глобальный минимум $f_3^{\min} = 0$ достигается в точке (0, 0) (то есть при $x_1 = 0$ и $x_2 = 0$ ).

В каждом эксперименте по поиску глобального оптимума тестовой функции использовались параметры базового PSO-алгоритма, PSO-GS-алгоритма и PSO-DOE-алгоритма, приведенные в таблице 2.

Таблица 2 – Параметры алгоритмов

Размерность пространства поиска	2
Размер популяции частиц	600
Максимальное количество поколений	1000
Коэффициент личного ускорения	2
Коэффициент глобального ускорения	5
Масштабирующий коэффициент	0.3

*Доля успешных запусков* – отношение числа успешных запусков, в которых алгоритм оптимизации обнаруживает заранее известный оптимум с требуемой точностью, к общему числу статистически независимых запусков, выраженное в процентах.

*Среднее время сходимости* – время, выраженное в секундах, измеряемое от начала работы (запуска) алгоритма оптимизации до момента первого нахождения глобально-оптимального решения с заданной точностью  $\varepsilon$ , усредненное по статистически независимым запускам.

*Средняя скорость сходимости* – количество поколений алгоритма оптимизации, при котором впервые было получено глобально-оптимальное решение с заданной точностью  $\varepsilon$ , усредненное по статистически независимым запускам.

Кроме того, были использованы такие показатели качества, как: *среднее значение целевой функции, дисперсия среднего времени сходимости, стандартное отклонение среднего времени сходимости* алгоритма оптимизации.

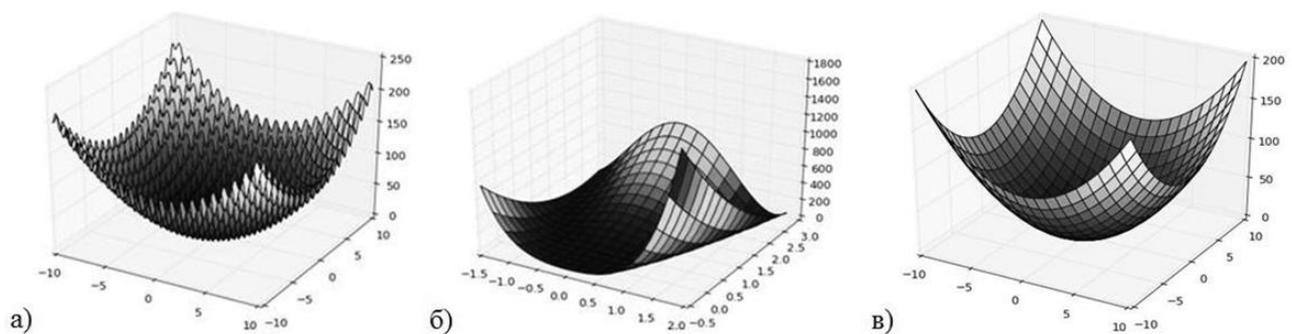


Рисунок 3 – Графики тестовых функций: а) функция Растринга, б) функция Розенброка, в) функция Сферы

При всех запусках всех алгоритмов оптимизации инициализация начальных популяций частиц (решений) осуществлялась с помощью идентичных (одинаковых) случайно сгенерированных начальных приближений решений задачи оптимизации. При этом расчеты с использованием гибридных версий PSO-алгоритма выполня-

лись при разном суммарном количестве узлов  $\gamma$  поиска по сетке (разном суммарном количестве вычислений значений целевой функции в узлах сетки), которое вычислялось для PSO-GS-алгоритма и PSO-DOE-алгоритма соответственно по формулам:

$$\gamma = (r + 1)^2, \tag{14}$$

$$\gamma = 13 \cdot z, \quad (15)$$

где  $r$  – количество интервалов разбиений на каждом  $j$ -м диапазоне поиска по сетке  $[d_{\min}^j, d_{\max}^j]$  ( $j=1,2$ );  $z$  – количество итераций DOE-алгоритма.

Результаты экспериментальных исследований показали, что оптимальными значениями параметров  $r$  и  $z$ , при которых достигаются лучшие значения показателей качества (минимальное значение среднего времени сходимости, минимальное значение средней скорости сходимости, минимальное среднее значение целевой функции, максимальная доля успешных запусков) для PSO-GS-алгоритма и PSO-DOE-алгоритма соответственно при оптимизации пред-

ставленных в таблице 1 целевых функций, являются:  $r = 10$  и  $z = 50$ .

При  $r = 5$  в PSO-GS-алгоритме поиск по сетке не улучшает показатели качества PSO-алгоритма, а при  $z = 5$  в PSO-DOE-алгоритме поиск по сетке улучшает результаты PSO-алгоритма только в его первых поколениях.

Графические зависимости, представленные на рисунке 4, позволяют определить оптимальное количество вычислений по сетке в GS-алгоритме и DOE-алгоритме на примере функции Сферы. Как видно из рисунка 4 минимальные значения среднего времени сходимости для PSO-GS-алгоритма и PSO-DOE-алгоритма достигаются соответственно при  $r = 10$  и  $z = 50$ .

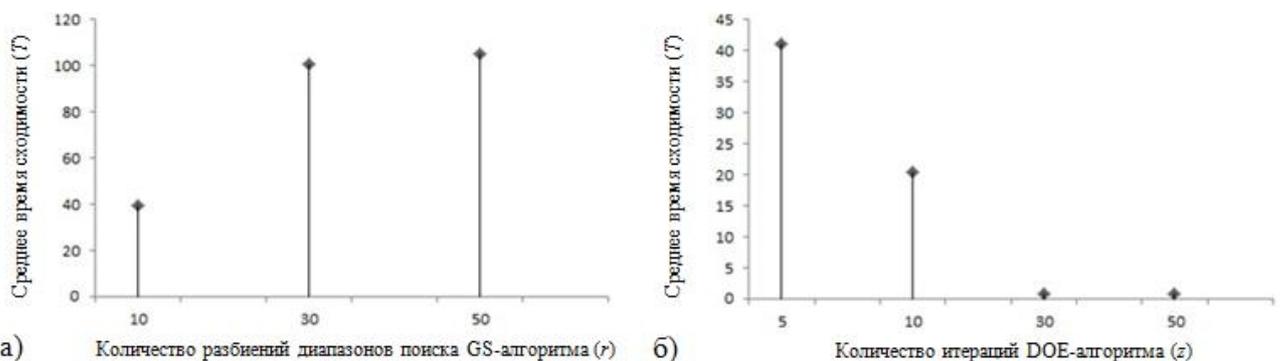


Рисунок 4 – Определение оптимального количества вычислений по сетке при реализации гибридных версий PSO-алгоритма на основе: а) GS-алгоритма, б) DOE-алгоритма

В таблице 3 приведены результаты экспериментальных расчетов. При этом жирным шрифтом выделена информация об алгоритмах оптимизации с лучшими значениями показателей качества.

Результаты экспериментальных расчетов,

приведенные в таблице 3, позволяют сделать вывод, что базовый PSO-алгоритм характеризуется худшими значениями показателей качества по сравнению с PSO-GS-алгоритмом и PSO-DOE-алгоритмом.

Таблица 3 – Результаты экспериментальных расчетов

Тестовая функция	Алгоритм	Количество вычислений в узлах сетки	Среднее время сходимости (T), сек.	Дисперсия величины T, сек <sup>2</sup> .	Стандартное отклонение величины T, сек.	Средняя скорость сходимости	Доля успешных запусков, %	Среднее значение целевой функции
$f_1$	PSO-GS	121 ( $r = 10$ )	39.41	808.05	28.58	316	94	0.00054
	<b>PSO-DOE</b>	<b>650</b> ( $z = 50$ )	<b>0.66</b>	<b>0.06</b>	<b>0.24</b>	<b>1</b>	<b>100</b>	<b>0</b>
	PSO	–	54.99	1516.08	39.14	330	99	0.00048
$f_2$	PSO-GS	121 ( $r = 10$ )	46.48	935.84	30.77	342	85	0.00055
	<b>PSO-DOE</b>	<b>650</b> ( $z = 50$ )	<b>40.75</b>	<b>1145.18</b>	<b>34.01</b>	<b>74</b>	<b>100</b>	<b>0.00073</b>
	PSO	–	53.64	1804.93	42.76	366	79	0.00047
$f_3$	PSO-GS	121 ( $r = 10$ )	39.55	650.08	25.63	295	94	0.00055
	<b>PSO-DOE</b>	<b>650</b> ( $z = 50$ )	<b>0.76</b>	<b>0.01</b>	<b>0.11</b>	<b>1</b>	<b>100</b>	<b>2.0968E-27</b>
	PSO	–	63.61	2720.81	52.44	341	94	0.00049

При этом PSO-DOE-алгоритм по сравнению с PSO-GS-алгоритмом позволяет находить глобальный оптимум тестовых функций в среднем за меньшее время, а также при его реализации обеспечивается большая доля успешных запусков и достигается меньшая погрешность вычисления значений глобального оптимума целевых функций (например, в случае функции Растригина и функции Сферы при использовании диапазонов поиска из таблицы 1).

Таким образом, в целом PSO-DOE-алгоритм характеризуется лучшими значениями показателей качества, приведенных в таблице 3. При этом лучшие значения показателей качества для PSO-DOE-алгоритма (по сравнению с другими алгоритмами оптимизации) достигались на различных диапазонах поиска: как на диапазонах поиска, приведенных в таблице 1, так и на меньших диапазонах.

Экспериментальным путем была подтверждена целесообразность применения PSO-GS-алгоритма и PSO-DOE-алгоритма для решения задачи подбора оптимальных значений параметров SVM-классификатора. При этом исследования проводились на реальных наборах данных, заимствованных из проекта Statlog [2] и библиотеки машинного обучения UCI [2]. В эксперименте были использованы два набора данных из медицинской диагностики (наборы WDBC и Heart) и набор данных для кредитного скоринга (набор Australian) [2]. Для всех наборов данных выполнялась бинарная классификация.

В качестве функции ядра использовалась радиальная базисная функция, вследствие чего решалась задача поиска оптимальных значений двух параметров – параметра регуляризации  $C$  и коэффициента  $\sigma$  функции радиального ядра.

С целью обеспечения объективного сравнения результатов экспериментов запуск базового PSO-алгоритма и предлагаемых PSO-GS-алгоритма и PSO-DOE-алгоритма для конкретного набора данных инициализировался идентичными случайно сгенерированными начальными популяциями частиц. Кроме того, использовались идентичные случайные разбиения исходного набора данных на обучающую и тестовую выборки данных.

Значения параметров SVM-классификатора полагались оптимальными, если они обеспечивали высокую точность классификации и минимальное количество опорных векторов на обучающей выборке. При этом значение показателя точности классификации  $OSR$  (Overall Success Rate) [2,7] вычислялось по формуле:

$$OSR = \frac{TP + TN}{TP + TN + FP + FN} \cdot 100\%, \quad (16)$$

где  $TP$  – количество истинно положительных наблюдений;  $TN$  – количество истинно отрицательных наблюдений;  $FP$  – количество ложноположительных наблюдений («ложных обнаружений», ошибка II рода);  $FN$  – количество ложноотрицательных наблюдений («ложных пропусков», ошибка I рода).

Во избежание недообучения и переобучения SVM-классификатора полагалось, что высокая точность классификации достигается в случае, если количество ошибок на обучающей и тестовой выборках данных минимально, при этом количество ошибок SVM-классификатора на обучающей и тестовой выборках данных практически не отличаются [2].

Для всех запусков алгоритмов оптимизации были установлены одинаковые значения параметров PSO-алгоритма и одинаковые диапазоны поиска значений искомых параметров SVM-классификатора. При этом значения параметров PSO-алгоритма соответствовали значениям параметров из таблицы 2, кроме значения параметра, определяющего максимальное количество поколений, равного в данном случае 20.

При этом в экспериментах № 1 и № 2 использовались соответственно следующие диапазоны поиска значений параметров  $C$  и  $\sigma$ :

- $0.1 \leq C \leq 10$  и  $0.1 \leq \sigma \leq 10$ ;
- $0.01 \leq C \leq 20$  и  $0.01 \leq \sigma \leq 20$ .

В таблице 4 представлены результаты экспериментальных расчетов по подбору оптимальных значений параметров SVM-классификатора с применением базового PSO-алгоритма, PSO-GS-алгоритма и PSO-DOE-алгоритма (при этом жирным шрифтом выделены алгоритмы с лучшими значениями показателей качества). В первом столбце таблицы 4 отражена информация о размере каждого исходного набора данных: количество объектов и количество характеристик. Например, для набора WDBC количество объектов равно 569, а количество характеристик – 30.

Экспериментально было установлено, что оптимальными значениями параметров  $r$  и  $z$  для PSO-GS-алгоритма и PSO-DOE-алгоритма являются соответственно: для эксперимента № 1 –  $r = 5$  и  $z = 5$ ; для эксперимента № 2 –  $r = 10$  и  $z = 5$  (для набора данных WDBC),  $r = 30$  и  $z = 10$  (для наборов данных Heart и Australian). На основе данных таблицы 4 можно сделать вывод, что PSO-GS-алгоритм и PSO-DOE-алгоритм решают задачу поиска оптимальных значений

параметров SVM-классификатора эффективнее, чем базовый PSO-алгоритм.

Таблица 4– Результаты экспериментальных расчетов

Набор данных	Эксперимент	Тип алгоритма	Найденные значения параметров		Ошибки		Число опорных векторов	Точность (%)	Время поиска (сек.)
			$C$	$\sigma$	при обучении	при тесте			
WDBC 569×30	1	PSO	7,54	0,10	1 из 427	4 из 142	176	99,12	6305
		PSO-DOE	4,56	0,10	1 из 427	4 из 142	177	99,12	2267
		<b>PSO-GS</b>	<b>7,50</b>	<b>0,10</b>	<b>1 из 427</b>	<b>4 из 142</b>	<b>176</b>	<b>99,12</b>	<b>788</b>
	2	PSO	4,48	0,04	3 из 427	2 из 142	91	99,12	1338,36
		<b>PSO-DOE</b>	<b>20</b>	<b>0,01</b>	<b>3 из 427</b>	<b>2 из 142</b>	<b>53</b>	<b>99,12</b>	<b>1088,02</b>
		PSO-GS	14,27	0,01	3 из 427	2 из 142	57	99,12	2895,76
Heart 270×13	1	PSO	2,65	0,11	2 из 192	14 из 78	126	94,07	304
		<b>PSO-DOE</b>	<b>3,13</b>	<b>0,10</b>	<b>2 из 192</b>	<b>13 из 78</b>	<b>122</b>	<b>94,44</b>	<b>160</b>
		PSO-GS	3,00	0,11	2 из 192	13 из 78	122	94,44	432
	2	PSO	3,20	0,08	6 из 192	13 из 78	112	92,96	230,76
		<b>PSO-DOE</b>	<b>7,00</b>	<b>0,07</b>	<b>1 из 192</b>	<b>14 из 78</b>	<b>104</b>	<b>94,44</b>	<b>822,16</b>
		PSO-GS	3,23	0,1	2 из 192	14 из 78	122	94,07	1022,45
Australian 690×14	1	PSO	2,67	0,11	31 из 492	22 из 198	253	92,32	5165
		PSO-DOE	3,71	0,10	28 из 492	21 из 198	246	92,90	2825
		<b>PSO-GS</b>	<b>3,83</b>	<b>0,10</b>	<b>29 из 492</b>	<b>21 из 198</b>	<b>246</b>	<b>92,75</b>	<b>978</b>
	2	PSO	3,56	0,11	27 из 492	21 из 198	255	93,04	2102
		<b>PSO-DOE</b>	<b>6,39</b>	<b>0,10</b>	<b>23 из 492</b>	<b>23 из 198</b>	<b>250</b>	<b>93,33</b>	<b>1852</b>
		PSO-GS	2,90	0,13	26 из 492	21 из 198	274	93,19	1636

По результатам эксперимента № 1 PSO-GS-алгоритм на наборах данных WDBC и Australian характеризовался меньшим временем поиска оптимальных значений параметров SVM-классификатора по сравнению PSO-DOE-алгоритмом, однако значения показателя точности классификации (16) практически не отличались для обоих алгоритмов.

По результатам эксперимента № 2 PSO-DOE-алгоритм оказался лучшим на всех представленных наборах данных по значениям показателя точности классификации  $OSR$  (который должен быть максимизирован) и по количеству опорных векторов  $SV$  (которое должно быть минимизировано). По времени поиска оптимальных значений параметров SVM-классификатора PSO-DOE-алгоритм практически не отличался от PSO-GS-алгоритма (на наборе данных Australian) или был лучше (на наборах данных WDBC и Heart).

Таким образом, при решении задачи поиска оптимальных значений параметров SVM-классификатора PSO-DOE-алгоритм характеризуется лучшими значениями показателей качества при достаточно широких диапазонах поиска иско-

мых значений параметров.

Очевидно, что при решении задачи оптимизации в отсутствии информации о значении глобально оптимального решения и необходимости использования широкого диапазона поиска целесообразно применять гибридный PSO-DOE-алгоритм.

### Заключение

Результаты экспериментальных исследований, полученные при решении задачи поиска экстремума тестовых функций, а также при решении задачи поиска оптимальных значений параметров SVM-классификатора, подтверждают целесообразность использования предлагаемых гибридных алгоритмов с целью повышения эффективности одного из популярных алгоритмов эволюционных вычислений – PSO-алгоритма.

Анализ эффективности PSO-GS-алгоритма и PSO-DOE-алгоритма при решении ряда задач оптимизации показал, что PSO-DOE-алгоритм в большинстве случаев характеризуется лучшими значениями показателей качества (такими, как среднее время сходимости, средняя скорость

сходимости и т.п.).

Необходимо отметить, что важной задачей при реализации базового PSO-алгоритма и его гибридных версий является настройка их параметров, что в значительной мере влияет на эффективность данных алгоритмов и требует исследований в каждом конкретном случае.

В дальнейшем предполагается модифицировать предложенные гибридные версии PSO-алгоритма для обеспечения возможности решения задачи многоцелевой оптимизации, в частности, с целью решения задачи разработки эффективного SVM-классификатора.

#### Библиографический список

1. Блюмин С. Л., Сытник К. И. Гибридизация оптимизационных методов эволюционных вычислений // Системы управления и информационные технологии. 2015. № 3 (61). С. 8-13.
2. Демидова Л. А., Соколова Ю. С. Аспекты применения алгоритма роя частиц в задаче разработки SVM-классификатора // Вестник Рязанского государственного радиотехнического университета. 2015. № 53. С. 84-92.
3. Демидова Л. А., Никульчев Е. В., Соколова Ю. С. Классификация больших данных: использование SVM-ансамблей и SVM-классификаторов с модифицированным роевым алгоритмом // Cloud of Science. 2016. Т. 3. № 1. С. 5-42.
4. Демидова Л. А., Соколова Ю. С. Разработка ансамбля SVM-классификаторов с использованием декорреляционного алгоритма максимизации // Информатика и системы управления. 2016. № 1 (47). С. 95-105.
5. Клюева И. А. Подбор параметров SVM-алгоритма с помощью алгоритма PSOGridSearch в задаче классификации рисков ИТ-проектов // Новые информационные технологии в научных исследованиях. Рязань: Рязанский государственный радиотехнический университет, 2015. С. 266-268.
6. Клюева И. А. Модификация алгоритма PSO-GS для подбора параметров SVM-алгоритма в задаче классификации рисков ИТ-проектов // Интеллектуальные и информационные системы. Тула: Тульский государственный университет, 2015. С. 62-65.
7. Клюева И.А. Гибридный алгоритм настройки параметров интеллектуального классификатора данных // Математические методы в технике и технологиях. Саратов: Саратовский государственный технический университет, 2015. Т. 7. С. 234-238.
8. Клюева И. А. Подходы к модификации алгоритма роя частиц // Информационные технологии в процессе подготовки современного специалиста. Липецк: ФГБОУ ВПО «ЛГПУ», 2015. Вып. 19. С. 40-46.
9. Demidova L., Nikulchev E., Sokolova Yu. The SVM Classifier Based on the Modified Particle Swarm Optimization // International Journal of Advanced Computer Science and Applications (IJACSA). 2016. Vol. 7. No. 2. pp. 16-24.
10. Jun Sun, Choi-Hong Lai, Xiao-Jun Wu. Particle Swarm Optimisation: Classical and Quantum Perspectives. CRC Press, 2011. 419 p.
11. Bio-Inspired Credit Risk Analysis. Computational Intelligence with Support Vector Machines / Ed. by Lean Yu, Shouyang Wang, Kin Keung Lai, Ligang Zhou. Springer-Verlag Berlin Heidelberg, 2008. 244 p.
12. Vapnik V. Statistical Learning Theory. New York: John Wiley & Sons, 1998. 732 p.
13. Joachims T. A support vector method for multivariate performance measures // In Proceedings of the International Conference on Machine Learning (ICML), 2005. pp. 377-384.
14. Карпенко А. П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой. М.: Изд-во МГТУ им. Н. Э. Баумана, 2014. 446 с.
15. Курейчик В. М., Кажаров А. А. Использование роевого интеллекта в решении NP-трудных задач // Известия Южного федерального университета. Технические науки. 2011. № 7 (120). С. 30-36.

UDC 004.855.5

## THE DEVELOPMENT AND STUDY OF HYBRID VERSIONS OF PARTICLE SWARM ALGORITHM BASED ON GRID SEARCH ALGORITHMS

L. A. Demidova, PhD (technical sciences), full professor, RSREU, Ryazan; liliya.demidova@rambler.ru  
I. A. Klyueva, post-graduate student, RSREU, Ryazan; i.aleschenko@yandex.ru.

*The present paper considers the approach to the problem solution of unconstrained optimization based on the hybridization of particle swarm optimization algorithm (PSO-algorithm) and grid search algorithm. The aim of this work is the development of hybrid versions of PSO-algorithm and study of its search characteristics. The paper presents two hybrid versions of basic PSO algorithm, involving the use correspondingly of classical Grid Search (GS) algorithm and Design of Experiment (DOE) algorithm. It is proposed to use*

canonical PSO-algorithm as base algorithm. The results of experimental studies confirming the application efficiency of the proposed hybrid versions of basic PSO-algorithm in solving optimization problems are represented. Herewith the comparative analysis of main quality indicators of basic PSO-algorithm and its hybrid versions in the problem solution of finding the global optimum of several test functions has been carried out. In addition, the application's expediency of hybrid versions of basic PSO-algorithm in order to reduce the time expenditures for searching of optimum parameters' of SVM-classifier is shown.

**Key words:** particle swarm optimization algorithm; grid search algorithm; hybrid algorithm; test function; classification, SVM-classifier, optimization parameters, radial basis kernel function.

**DOI:** 10.21667/1995-4565-2016-57-3-105-116

### References

1. **Bljumin S. L., Sytnik K. I.** Gibridizacija optimizacionnyh metodov jevoljucionnyh vychislenij. *Sistemy upravlenija i informacionnye tehnologii*. 2015, no. 3 (61), pp. 8-13 (in Russian).

2. **Demidova L. A., Sokolova Yu. S.** Aspekty primeneniya algoritma roja chastic v zadache razrabotki SVM-klassifikatora. *Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta*. 2015, no. 53, pp. 84-92 (in Russian).

3. **Demidova L. A., Nikul'chev E. V., Sokolova Yu. S.** Klassifikacija bol'shih dannyh: ispol'zovanie SVM-ansamblej i SVM-klassifikatorov s modifitsirovannym roevym algoritmom. *Cloud of Science*. 2016, vol. 3, no. 1, pp. 5-42 (in Russian).

4. **Demidova L. A., Sokolova Yu. S.** Razrabotka ansamblja SVM-klassifikatorov s ispol'zovaniem dekorreljacionnogo algoritma maksimizacii. *Informatika i sistemy upravlenija*. 2016, no. 1 (47), pp. 95-105 (in Russian).

5. **Klyueva I. A.** Podbor parametrov SVM-algoritma s pomoshh'ju algoritma PSOGridSearch v zadache klassifikacii riskov IT-proektov (The parameter's selection of the SVM-algorithm by means of the algorithm PSOGridSearch in the classification problem of IT-project's risks). *Novye informacionnye tehnologii v nauchnyh issledovanijah*. Rjazan': Rjazanskij gosudarstvennyj radiotekhnicheskij universitet, 2015, pp. 266-268 (in Russian).

6. **Klyueva I. A.** Modifikacija algoritma PSO-GS dlja podbora parametrov SVM-algoritma v zadache klassifikacii riskov IT-proektov (The modification of the algorithm PSO-GS for the searching of the parameters of the SVM-algorithm in the classification problem of IT-project's risks). *Intellektual'nye i informacionnye sistemy*. Tula: Tul'skij gosudarstvennyj universitet, 2015, pp. 62-65 (in Russian).

7. **Klyueva I. A.** Gibridnyj algoritm nastrojki parametrov intellektual'nogo klassifikatora dannyh (The

parameter's setting hybrid algorithm of intellectual data classifier). *Matematicheskie metody v tehnike i tehnologijah*. Saratov: Saratovskij gosudarstvennyj tekhnicheskij universitet, 2015, vol. 7, pp. 234-238 (in Russian).

8. **Klyueva I. A.** Podhody k modifikacii algoritma roja chastic (The approaches to modification of the particle swarm optimization algorithm). *Informacionnye tehnologii v processe podgotovki sovremennogo specialisty*. Lipeck: FGBOU VPO «LGPU», 2015, issue 19, pp. 40-46 (in Russian).

9. **Demidova L., Nikulchev E., Sokolova Yu.** The SVM Classifier Based on the Modified Particle Swarm Optimization. *International Journal of Advanced Computer Science and Applications (IJACSA)*. 2016, vol. 7, no. 2, pp. 16-24.

10. **Jun Sun, Choi-Hong Lai, Xiao-Jun Wu.** *Particle Swarm Optimisation: Classical and Quantum Perspectives*, CRC Press, 2011, 419 p.

11. *Bio-Inspired Credit Risk Analysis. Computational Intelligence with Support Vector Machines* / Ed. by **Lean Yu, Shouyang Wang, Kin Keung Lai, Ligang Zhou**, Springer-Verlag Berlin Heidelberg, 2008, 244 p.

12. **Vapnik V.** *Statistical Learning Theory*, New York, John Wiley & Sons, 1998, 732 p.

13. **Joachims T.** A support vector method for multivariate performance measures. *In Proceedings of the International Conference on Machine Learning (ICML)*, 2005, pp. 377-384.

14. **Karpenko A. P.** *Sovremennye algoritmy poiskovoj optimizacii. Algoritmy, vdohnovlennye prirodoj* (The modern algorithms of the search optimization. The algorithms inspired by nature), Moscow, Izd-vo MGTU im. N. Je. Bauman, 2014, 446 p. (in Russian).

15. **Kurejchik V. M., Kazharov A. A.** Ispol'zovanie roevogo intellekta v reshenii NP-trudnyh zadach. *Izvestija Juzhnogo federal'nogo universiteta. Tehnicheskie nauki*. 2011, no. 7 (120), pp. 30-36 (in Russian).