Разработка SVM-классификатора с применением гибридных версий алгоритма роя частиц на основе поиска по сетке

Л. А. Демидова, И. А. Клюева

Рязанский государственный радиотехнический университет
390005, Рязань, ул. Гагарина, 59/1

e-mail: liliya.demidova@rambler.ru, i.aleschenko@yandex.ru

Аннотация. Рассматриваются подходы к решению задачи поиска параметров SVM-классификатора на основе гибридизации алгоритма роя частиц (PSO-алгоритм) и алгоритмов поиска по сетке с целью обеспечения высокого качества классификационных решений. В работе представлены две гибридные версии базового PSO-алгоритма, предполагающие использование соответственно классического алгоритма Grid Search (GS-алгоритм) и алгоритма Design of Experiment (DOE-алгоритм). При этом в качестве базового используется канонический PSO-алгоритм. Результаты проведенных исследований демонстрируют целесообразность применения гибридных версий базового PSO-алгоритма с целью сокращения временных затрат на поиск оптимальных значений параметров SVM-классификатора при сохранении высокого качества его классификационных решений.
Ключевые слова: классификация, показатели качества классификации, гибридизация, алгоритм роя частиц, алгоритм поиска по сетке, SVM-классификатор, параметр регуляризации, радиально базисная функция ядра.

1. Введение

Классификация данных — одна из наиболее распространенных задач машинного обучения (*machine learning*) [1–9]. Для решения этой задачи требуется создание классифицирующей функции, которая присваивает каждому набору входных данных значение метки одного из классов. Классификация новых данных производится после прохождения этапа «обучения», в процессе которого на вход обучающего алгоритма подаются данные с уже приписанными им метками классов.

В настоящее время для решения широкого спектра классификационных задач в различных прикладных областях успешно применяется SVM-алгоритм (*Support Vector Machine*, SVM) [1–9], являющийся алгоритмом машинного обучения по прецедентам. SVM-алгоритм реализует построение бинарного SVM-классификатора.

SVM-алгоритм реализует построение разделяющей гиперплоскости, которая разделяет объекты с разной классовой принадлежностью. При этом по обеим сто-

ронам разделяющей гиперплоскости строятся две параллельные гиперплоскости, задающие границы классов и находящиеся на максимально возможном расстоянии друг от друга. Предполагается, что чем больше расстояние между этими параллельными гиперплоскостями, тем меньше средняя ошибка SVM-классификатора. Векторы характеристик классифицируемых данных, ближайшие к параллельным гиперплоскостям, называются опорными векторами.

При классификации реальных наборов данных в большинстве случаев отсутствует возможность линейной разделимости объектов на классы. В связи с этим главной особенностью SVM-классификатора в случае нелинейной разделимости объектов является применение специальной функции, называемой ядром, используемой для перевода экспериментального набора данных из исходного пространства характеристик в пространство более высокой размерности, в котором строится гиперплоскость, разделяющая классы. В процессе обучения SVM-алгоритма одной из приоритетных задач является настройка параметров SVM-классификатора, наиболее важными из которых являются тип функции ядра, значения параметров ядра и значение параметра регуляризации.

В качестве функции ядра, позволяющей разделить объекты разных классов, обычно используется одна из следующих функций [1, 3]: линейная, полиномиальная, радиальная базисная, сигмоидная.

Параметр регуляризации C позволяет найти компромисс между максимизацией ширины полосы, разделяющей классы, и минимизацией суммарной ошибки. Другими словами, параметр регуляризации контролирует соотношение между гладкой границей и корректной классификацией рассматриваемых данных.

В случае применения радиальной базисной функции ядра (*Radial Basis Function,* RBF) [3] необходимо определить значение коэффициента данной функции σ.

Простейший подход к настройке параметров SVM-классификатора основан на простом переборе различных комбинаций значений параметров. Наиболее часто с целью настройки параметров SVM-классификатора применяются алгоритмы поиска по сетке, в частности алгоритм *Grid Search* (GS-алгоритм) [5]. При этом для каждой комбинации параметров, соответствующей определенному узлу сетки, осуществляется перекрестная проверка (кросс-валидация, *Cross-validation*) [5] на обучающем наборе данных. В результате выбирается комбинация значений параметров, определяющая некоторый узел сетки и характеризующаяся лучшим значением показателя кросс-проверки.

Нахождение оптимального набора значений параметров SVM-классификатора позволяет избежать проблемы переобучения (маленькая ошибка обучающей выборке, большая ошибка на тестовой выборке) или проблемы недообучения (ошибки

на обучающей и тестовой выборках близки между собой и являются большими по величине) классификатора. Если ошибки на обучающей и тестовой выборках близки между собой и невелики по значению, то такой SVM-классификатор признается искомым для решения задачи классификации.

Поскольку при построении классификаторов используется сложная, многоэкстремальная и многопараметрическая целевая функция, целесообразно применять поиск ее оптимума сразу по всему пространству возможных решений.

В настоящее время широкое применение находят алгоритмы оптимизации, созданные по образу существующих в природе биологических систем. К таким алгоритмам относятся биоинспирированные алгоритмы стохастической оптимизации: генетический алгоритм, алгоритм роя частиц, муравьиный алгоритм, пчелиный алгоритм [8]. Данные алгоритмы оперирует множествами простых сущностей на всем пространстве поиска, моделируя интеллектуальное поведение популяции, в которой каждая особь представляет некоторое альтернативное приближенное решение.

В последние годы все большее применение при решении различных прикладных задач оптимизации находит алгоритм роя частиц (*Particle Swarm Optimization*, PSO-алгоритм) [1–3, 7–16], основанный на идее о возможности решения задач оптимизации посредством моделирования поведения групп животных.

PSO-алгоритм характеризуется простотой реализации и, вследствие этого, низкой алгоритмической сложностью, поскольку для его реализации достаточно определить только значение оптимизируемой функции. В связи с этим можно сделать вывод о целесообразности применения PSO-алгоритма к решению задачи поиска оптимальных значений параметров SVM-классификатора.

В настоящее время известны различные способы повышения эффективности базового PSO-алгоритма, которые можно разделить на метаоптимизационные и комбинационные [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. Предполагается использовать два алгоритма поиска по сетке: классический алгоритм *Grid Search* (GS-алгоритм) и алгоритм *Design of Experiment* (DOE-алгоритм) [3, 5, 8].

Цель работы заключается в разработке гибридных версий базового PSO-алгоритма на основе алгоритмов поиска по сетке и сравнение их поисковых характеристик. В рамках решения задачи поиска оптимальных значений параметров SVM-классификатора планируется выполнить тестирование разработанных гибридных версий PSO-алгоритма на реальных наборах данных. Основными показателями оценки эффективности реализованных алгоритмов являются время поиска оптимальных значений параметров SVM-классификатора, показатели качества

классификации данных (общая точность, чувствительность, специфичность, количество опорных векторов). При этом рассматривается задача бинарной классификации.

2. Принципы реализации SVM-алгоритма

В результате обучения SVM-классификатора определяется разделяющая гиперплоскость (рис. 1) [3], которая может быть задана уравнением $\langle w, z\rangle + b = 0$, где w — вектор-перпендикуляр к разделяющей гиперплоскости; b — параметр, соответствующий кратчайшему расстоянию от начала координат до гиперплоскости; $\langle w, z\rangle$ — скалярное произведение векторов w и z. Условие $-1 < \langle w, z\rangle + b < 1$ задает полосу, которая разделяет классы. Чем шире эта полоса, тем увереннее можно классифицировать объекты. Объекты, ближайшие к разделяющей гиперплоскости и расположенные на границах полосы, разделяющей классы, называются опорными векторами. Они несут основную информацию о разделении классов.

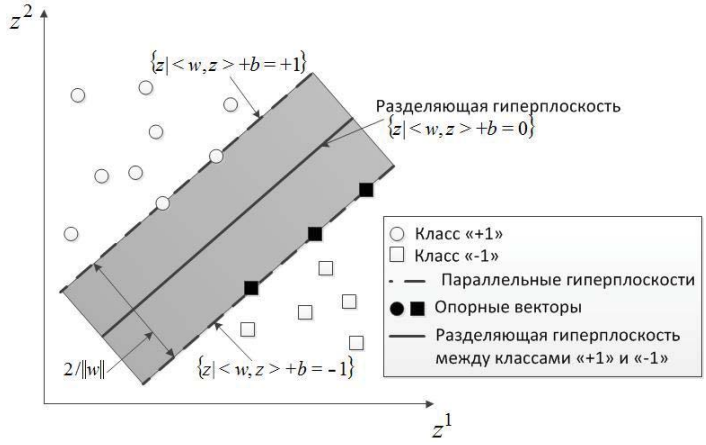


Рисунок 1. Построение разделяющей гиперплоскости в пространстве D-2

В SVM-алгоритме в случае нелинейной разделимости объектов одной из основных задач является определение типа спрямляющей функции ядра (kernel function) и подбор оптимальных значений для некоторого набора параметров с целью построения эффективного SVM-классификатора.

Классификация некоторого объекта z может быть выполнена с использованием следующего правила [3]:

$$F(z) = \text{sign}\left(\sum_{i=1}^{L} \lambda_i y_i \kappa(z_i, z) + b\right), \quad (1)$$

где λ_i — двойственная переменная функции Лагранжа; z_i — объект из обучающей выборки; $y_i \in Y = \{-1; +1\}$ — число, характеризующее классовую принадлежность объекта z_i из обучающей выборки; $\kappa(z_i, z)$ — функция ядра; C — параметр регуляризации $(C > 0)$; L — количество объектов в обучающей выборке; $i = \overline{1, L}$.

Наиболее полное математическое описание SVM-алгоритма приведено в [1, 3].

Основная проблема, возникающая при обучении SVM-классификатора, связана с отсутствием рекомендаций по выбору значения параметра регуляризации C, функции, описывающей ядро $\kappa(z_i, z)$, а также значений параметров самой функции ядра, при которых обеспечивается высокая точность классификации данных. Эта проблема может быть решена с применением тех или иных оптимизационных алгоритмов, в частности с использованием PSO-алгоритма [1].

Для построения SVM-классификатора в случае нелинейной разделимости данных на классы часто применяется радиальная базисная функция ядра (Radial Basis Function, RBF) [3]:

$$\kappa(z_i, z) = \exp\left[-\|z_i - z\|^2 / (2\sigma^2)\right], \qquad (2)$$

где параметр $\sigma > 0$.

В этом случае при разработке SVM-классификатора, наряду со значением параметра регуляризации C, необходимо определить значение параметра σ радиальной базисной функции ядра.

3. Принципы реализации алгоритма роя частиц и его гибридных версий

В PSO-алгоритме пространство поиска заполняется популяцией частиц, каждая из которых характеризуется своим положением (координатами) в пространстве поиска и скоростью. Кроме того, каждая частица способна запоминать свое лучшее положение в рое, а также обмениваться с другими частицами информацией о глобально «лучшей» позиции среди всех частиц.

Для каждого положения частицы роя вычисляется соответствующее значение целевой функции, на основе которого по определенным правилам [8] вычисляют новое положение (координаты) и новую скорость частицы в пространстве поиска.

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

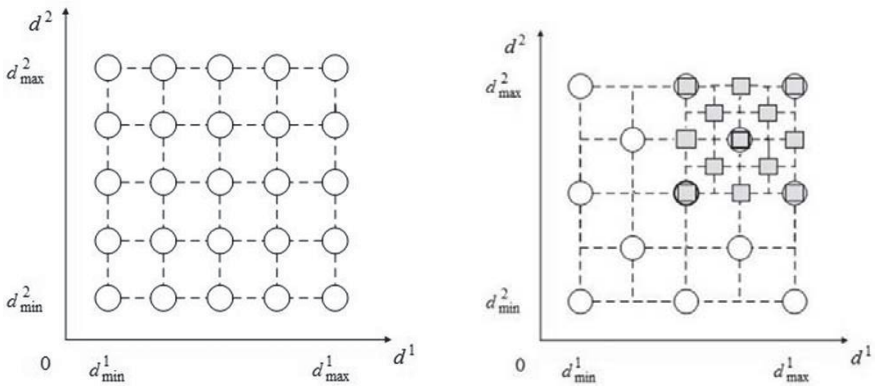
Основные принципы организации расчетов по вычислению нового положения и новой скорости частиц приведены в [8].

В настоящее время известны различные версии PSO-алгоритма. Традиционное применение получила одна из самых распространенных версий — каноническая, в которой предлагается выполнять нормировку коэффициентов ускорения, чтобы сходимость алгоритма не так сильно зависела от выбора их значений [8].

В последние годы все большее применение находят подходы, реализующие гибридизацию PSO-алгоритма с другими алгоритмами оптимизации с целью повышения эффективности классического PSO-алгоритма [8].

В данной работе представляется реализация двух гибридных версий PSO-алгоритма с применением двух алгоритмов поиска по сетке: классического GS-алгоритма и DOE-алгоритма [3, 5, 8].

Предлагаемые гибридные версии PSO-алгоритма разрабатывались в первую очередь для решения задачи поиска оптимальных значений параметров SVM-классификатора на основе радиальной базисной функции ядра. Данные алгоритмы, оперирующие набором частиц в пространстве поиска D-2, могут быть применены и для решения других оптимизационных задач соответствующей размерности, а также могут быть адаптированы на случай большей размерности пространства поиска.



а) *б)*

Рисунок 2. Формирование сетки в пространстве D-2:
а) в GS-алгоритме; б) в DOE-алгоритме

При создании гибридной версии PSO-алгоритма предлагается на каждой итерации PSO-алгоритма выполнять уточнение положения (т. е. 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, т. е. 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.

Поскольку в GS-алгоритме (рис. 2*а*) исследуются все узлы сетки, каждый из которых соответствует конкретной комбинации значений параметров оптимизации, преимущество данного алгоритма заключается в скрупулезности нахождения глобально-оптимального решения.

Достоинства альтернативного алгоритма поиска по сетке — DOE-алгоритма (рис. 2*б*), заключаются в следующем.

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.

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.

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

Шаг 1. Определить начальные характеристики частиц в рое (т. е. координаты и скорости частиц); инициализировать значения настраиваемых параметров PSO-алгоритма (число частиц в рое, количество итераций PSO-алгоритма, границы диапазонов поиска и др).

Шаг 2. Реализовать один шаг PSO-алгоритма. Скорректировать для каждой i-й частицы $(i = \overline{1, m})$; скорость $\vec{v}_i(t)$: $\vec{v}_i = (v_i^1, v_i^2, ..., v_i^n)$, и текущее положение (координаты) $\vec{x}_i(t)$: $\vec{x}_i = (x_i^1, x_i^2, \ldots, x_i^n)$, где n — размерность пространства поиска (т. е. n — количество параметров оптимизации), а m — количество частиц в рое. Найти координаты глобально «лучшей» частицы в рое (с лучшим значением целевой функции) и записать их в вектор $\vec{g}(t)$, предназначенный для хранения координат глобально «лучшей» частицы в рое, достигнутых популяцией частиц к текущему поколению. При условии, что целевая функция представляется как $f(x) = f(x^1, x^2, ..., x^n)$, под положением (координатами) глобально «лучшей» частицы в рое будем понимать точку пространства поиска, в которой по результатам всех итераций PSO-алгоритма, начиная с первой итерации до текущей, достигнуто

минимальное значение целевой функции в задаче поиска минимума функции: $f(x) \to \min\limits_{x \in R^n}$.

Шаг 3. Определить границы диапазонов поиска для одного из алгоритмов поиска по сетке (GS-алгоритма или DOE-алгоритма). При этом определить диапазоны $[d^j_{\min}, d^j_{\max}]$ $(j = \overline{1,n})$ с учетом диапазонов $[r^j_{\min}, r^j_{\max}]$ разброса частиц в рое на текущей итерации PSO-алгоритма. В качестве значений координат χ^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

$$l^j = \min\{\chi^j - r^j_{\min}, r^j_{\max} - \chi^j\}, \quad (3)$$

а границы диапазонов поиска по сетке определяются как:

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

$$d^j_{\max} = \chi^j + l^j. \quad (5)$$

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 «лучшей» 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 $[d^j_{\min}, d^j_{\max}]$ $(j = \overline{1,n})$, найденные на основе формул (4) и (5) на шаге 3 гибридной версии PSO-алгоритма, разбиваются на за-

данное 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 (т. е. 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 $[d^j_{\min}, d^j_{\max}]$ $(j = 1, 2)$ 13 узлов сетки (на рис. 2*б* 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 χ^j $(j = 1, 2)$ (пример на рис. 2*б* — маркер круглой формы с выделенным контуром) соответствует глобально «лучшей» частице роя, а ширина диапазонов поиска на текущей итерации DOE-алгоритма определяется как $S^j = d^j_{\max} - d^j_{\min}$ $(j = 1, 2)$.

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

$[\chi^1 - S^1/2, \chi^2 - S^2/2]$, $[\chi^1 - S^1/2, \chi^2 + S^2/2]$, $[\chi^1 + S^1/2, \chi^2 + S^2/2]$,
$[\chi^1 + S^1/2, \chi^2 - S^2/2]$, $[\chi^1 - S^1/2, \chi^2]$, $[\chi^1, \chi^2 + S^2/2]$, $[\chi^1 + S^1/2, \chi^2]$,
$[\chi^1, \chi^2 - S^2/2]$, $[\chi^1 - S^1/4, \chi^2 - S^2/4]$, $[\chi^1 - S^1/4, \chi^2 + S^2/4]$,
$[\chi^1 + S^1/4, \chi^2 + S^2/4]$, $[\chi^1 + S^1/4, \chi^2 - S^2/4]$, $[\chi^1, \chi^2]$.

Шаг 2. Вычислить значение целевой функции в каждом узле сетки и найти координаты φ^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:

$$d^j_{\min} = \varphi^j - S^j/2, \quad (6)$$

$$d^j_{\max} = \varphi^j + S^j/2. \quad (7)$$

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 χ^j $(j = 1, 2)$ (пример на рис. 2*б* — маркер квадратной формы с выделенным контуром) принимаются значения координат «лучшего» узла φ^j $(j = 1, 2)$, найденного на текущей итерации DOE-алгоритма.

Следует отметить, что границы диапазонов $[d^j_{\min}, d^j_{\max}]$ $(j = 1, 2)$ для первой итерации DOE-алгоритма вычисляются на основе формул (4) и (5) на шаге 3 гибридной версии PSO-алгоритма, а для всех остальных итераций DOE-алгоритма — на основе формул (6) и (7) 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 диапазонов поиска $[range^j_{\min}, range^j_{\max}]$ $(j = 1, 2)$. В случае если после вычисления по формулам (6) и (7) новых границ диапазонов поиска по сетке $[d^j_{\min}, d^j_{\max}]$ $(j = 1, 2)$ оказалось, что выполняется одно из условий $d^j_{\min} < range^j_{\min}$ при некотором $j = j^* \in \{1, 2\}$ или $d^j_{\max} > range^j_{\max}$ 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 $d^j_{\max} > range^j_{\max}$ при некотором $j = j^* \in \{1, 2\}$, то

$$d^{j^*}_{min} = \varphi^{j^*} - (\varphi^{j^*} - range^{j^*}_{min}), \quad (8)$$

$$d^{j^*}_{max} = \varphi^{j^*} + (\varphi^{j^*} - range^{j^*}_{min}); \quad (9)$$

если $d^j_{\max} > range^j_{\max}$ при некотором $j = j^* \in \{1, 2\}$, то

$$d^{j^*}_{min} = \varphi^{j^*} - (range^{j^*}_{max} - \varphi^{j^*}), \quad (10)$$

$$d^{j^*}_{max} = \varphi^{j^*} + (range^{j^*}_{max} - \varphi^{j^*}). \quad (11)$$

В результате реализации данной гибридной версии PSO-алгоритма производится поиск решения той или иной оптимизационной задачи.

4. Результаты экспериментальных исследований

Целесообразность использования предлагаемых гибридных алгоритмов подтверждается результатами решения нескольких оптимизационных задач. В частности, были рассмотрены такие задачи, как задача поиска оптимального глобального решения ряда тестовых функций и задача поиска оптимальных значений параметров SVM-классификатора.

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 алгоритмов проводилась с помощью высокоуровневого языка программирования Python (среда программирования Python 3.5). При этом использовался SVM-алгоритм из библиотеки машинного обучения Scikit-Learn. Реализация оптимизационных алгоритмов для тестовых функций. Сравнительный анализ указанных выше трех оптимизационных алгоритмов был выполнен в рамках решения задачи поиска глобального оптимума ряда тестовых функций. В частности, результаты экспериментальных исследований для целевых функций Растригина, Розенброка и функции сферы приведены в [8].

Полученные результаты [8] позволяют сделать вывод, что базовый PSO-алгоритм характеризуется худшими значениями показателей качества [8], такими как среднее время сходимости, средняя скорость сходимости, среднее значение целевой функции, доля успешных запусков, по сравнению с PSO-GS-алгоритмом и PSO-DOE-алгоритмом. 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 тестовых функций [8]. Реализация оптимизационных алгоритмов для настройки параметров SVM-классификатора. Экспериментальным путем была подтверждена перспективность применения PSO-GS-алгоритма и PSO-DOE-алгоритма для решения задачи подбора оптимальных значений параметров SVM-классификатора.

Исследования проводились на наборах данных, заимствованных из проекта Statlog и библиотеки машинного обучения UCI. Для всех наборов данных выполнялась бинарная классификация. В настоящей работе использовались следующие наборы данных (см. табл. 1):

- набор данных для медицинской диагностики болезни сердца – Heart (270 объектов, 13 характеристик; источник http:// archive.ics.uci.edu/ ml/machine-learning-databases/statlog/heart/);
- набор данных для кредитного скоринга о заявках на потребительские кредиты — Australian (690 объектов, 14 характеристик; источник http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/australian/);
- набор тестовых данных — МОТП12 (400 объектов, 2 характеристики; источник http://www.machinelearning.ru/wiki/index.php?title=Изображение:MOTP12_svm_example.rar).

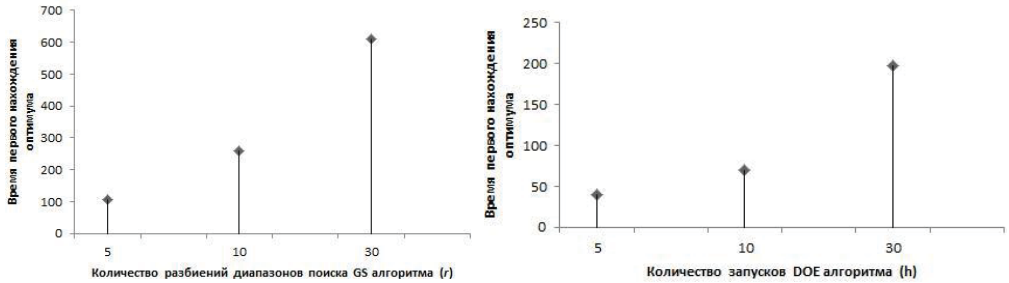
Расчеты с использованием гибридных версий PSO-алгоритма выполнялись при разном суммарном количестве γ узлов поиска по сетке (т. е. при разном суммарном количестве вычислений значений целевой функции в узлах сетки), которое вычислялось для PSO-GS-алгоритма и PSO-DOE-алгоритма соответственно по формулам

$$\gamma = (r+1)^2, \tag{12}$$

$$\gamma = 13 \cdot h, \tag{13}$$

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

Выбор оптимальных значений параметров SVM-классификатора производился по результатам нескольких экспериментов для разных значений параметров r и h (рис. 3, частный случай для выборки МОТП12). На основе критерия минимального значения времени первого нахождения оптимума в качестве оптимальных в настоящей работе выбраны следующие значения: $r = 5$ и $h = 5$.



а) *б)*

Рисунок 3. Определение оптимального количества вычислений по сетке при реализации гибридных версий PSO-алгоритма на основе: а) GS-алгоритма; б) DOE-алгоритма

При разработке SVM-классификатора использовалась радиальная базисная функция ядра (2). Вследствие чего PSO-алгоритм и его гибридные версии применялись для поиска оптимальных значений двух параметров SVM-классификатора: параметра регуляризации C и коэффициента функции ядра σ (т. е. расчеты выполнялись в пространстве поиска D-2). При этом изначально полагалось, что радиально базисная функция является априори оптимальной функцией ядра в контексте решаемой задачи классификации.

Значения параметров SVM-классификатора полагались оптимальными, если они обеспечивали высокую точность классификации и минимальное количество опорных векторов на обучающей выборке. Оценка качества классификации может быть выполнена с применением различных показателей качества классификации, среди которых следует выделить: показатель общей точности (Accuracy, Acc), называемый также показателем общего успеха (Overall Success Rate, OSR); показатель чувствительности (Sensitivity, Se), называемый также показателем полноты (Recall, Re); показатель специфичности (Specificity, Sp); показатель точности (Precision, Pr); а также показатель сбалансированной F-меры (F-measure, F1), которые вычисляются в соответствии с формулами

$$OSR = \frac{TP + TN}{TP + TN + FP + FN}, \tag{14}$$

$$Se = \frac{TP}{TP + FN}, \tag{15}$$

$$Sp = \frac{TN}{TN + FP}, \tag{16}$$

$$Pr = \frac{TP}{TP + FP}, \tag{17}$$

$$F1 = \frac{2 \cdot Pr \cdot Re}{Pr + Re}, \tag{18}$$

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); $Re = Se$.

Показатель общей точности OSR показывает долю истинно предсказанных наблюдений по отношению к общему числу наблюдений классификатора.

Показатель чувствительности Se показывает, какая доля от общего числа реальных положительных наблюдений предсказана в качестве положительных, т. е. говорит о том, насколько классификатор «пессимистичен» в своих оценках или как часто он «отбрасывает» (а это происходит при низком значении показателя Se)

наблюдения нужного класса. Этот показатель также называют показателем полноты *Re*.

Показатель специфичности *Sp* показывает, какая доля от общего числа реальных отрицательных наблюдений предсказана в качестве отрицательных.

Показатель точности *Pr* показывает, сколько из предсказанных положительных наблюдений являются действительно положительными, т. е. говорит о том, насколько классификатор оптимистичен в своих оценках или как часто он «предпочитает» (а это происходит при низком значении показателя *Pr*) присоединять наблюдения других классов к заданному.

Показатель сбалансированной F-меры (*F1*) вычисляет гармоническое среднее между показателем точности *Pr* и показателем полноты *Re*. При этом в формуле (18) этим показателям приписан одинаковый вес.

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 отличается [1, 8].

Для всех запусков алгоритмов оптимизации были установлены одинаковые значения параметров PSO-алгоритма и одинаковые диапазоны поиска значений искомых параметров SVM-классификатора.

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. При этом в процессе построения SVM-классификатора размер тестовой выборки составлял 20% от размера исходной выборки.

Для оценки качества бинарной классификации использовался ROC-анализ [3]. ROC-кривая, также известная как кривая ошибок, отображает соотношение между долей верных положительных классификаций от общего числа положительных классификаций (true positive rate — TPR) и долей ошибочных положительных классификаций от общего числа отрицательных классификаций (false positive rate — FPR). Показатель *AUC* (площадь под ROC-кривой) дает количественную интерпретацию ROC-кривой. Считается, что чем выше показатель *AUC*, тем качественнее классификатор.

На рис. 4 изображены ROC-кривые для SVM-классификаторов, построенные по данным тестовых выборок для трех описанных выше исходных наборов данных,

а также показатель *AUC* для каждого SVM-классификатора. Настройка параметров SVM-классификаторов производилась с использованием базового PSO-алгоритма и его гибридных версий.

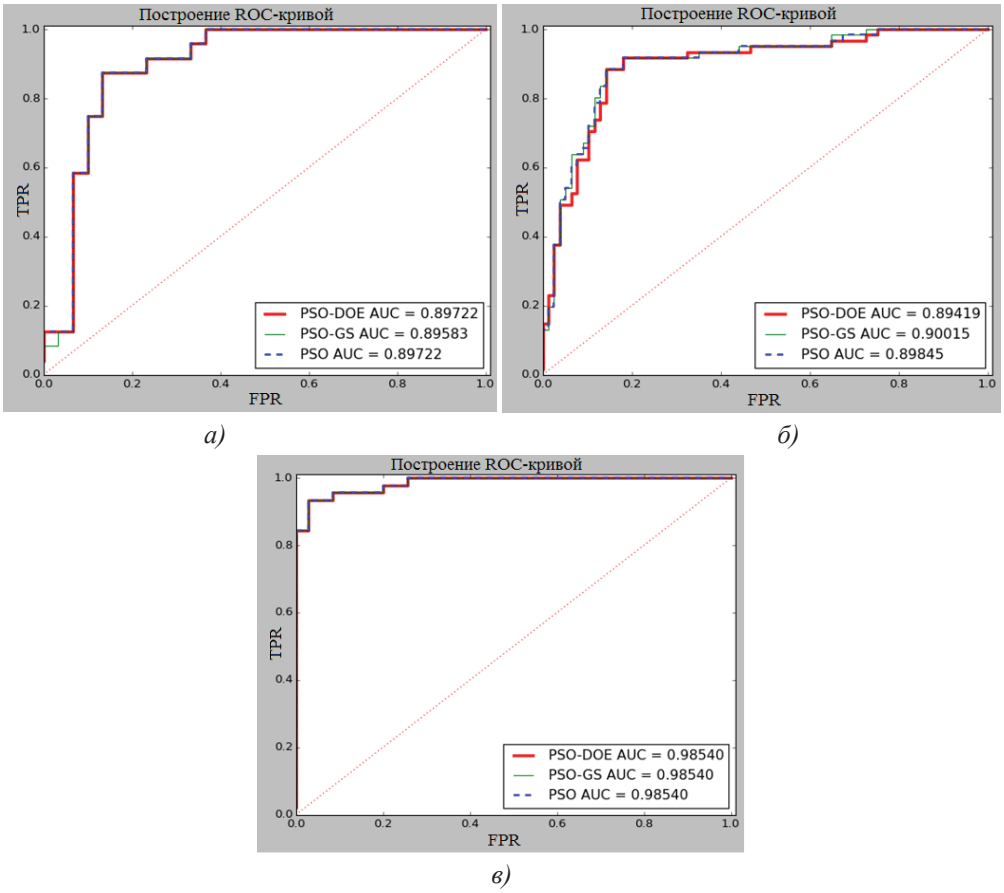


Рисунок 4. ROC-кривые для SVM-классификаторов, построенных с использованием базового PSO-алгоритма и его гибридных версий: а) для выборки Heart; б) для выборки Australian; в) для выборки МОТП12

Результаты ROC-анализа, в том числе и результаты сравнительного анализа значений показателя *AUC* показывают, что, на первый взгляд, расхождения SVM-классификаторов совершенно незначительны и сложно определить качество классификации. Однако при представлении результатов классификации наборов данных в виде таблицы 1, в которой отражено количество верно и ошибочно классифицируемых объектов, преимущество по качеству классификации следует отдать гибридным версиям базового PSO-алгоритма.

Таблица 1. Результаты реализации алгоритмов

Набор данных	Количество объектов	Количество признаков	Версия PSO алгоритма	Найденные параметры		Количество ошибок (класс «1»/ класс «–1»)		Количество опорных векторов	Точность (%)	Чувствительность (%)	Специфичность (%)	F-мера (*F*1)	Номер итерации первого нахождения оптимума	Время первого нахождения оптимума (сек.)	Время (общее) (сек.)
				C	σ	На обучающей выборке	На тестовой выборке								
Heart	270	13	Базов. PSO	8.87	0.05	6 (2/4)	7 (3/4)	108	95.19	96.67	93.33	0.9571	17	523	642
			PSO-GS	9.82	0.05	5 (1/4)	7 (3/4)	107	95.56	97.33	93.33	0.9605	9	362	714
			PSO-DOE	**9.98**	**0.05**	**5 (1/4)**	**7 (3/4)**	**107**	**95.56**	**97.33**	**93.33**	**0.9605**	**6**	**243**	**712**
Australian	690	14	Базов. PSO	9.48	0.13	11 (5/6)	18 (7/11)	276	95.80	96.09	95.56	0.9532	12	1546	2872
			PSO-GS	9.73	0.13	12 (5/7)	18 (7/11)	273	95.65	96.09	95.30	0.9516	5	1031	3481
			PSO-DOE	**9.99**	**0.13**	**10 (5/5)**	18 (7/11)	**273**	**95.95**	**96.09**	**95.82**	**0.9547**	**4**	**789**	**3292**
МОТП 12	400	2	Базов. PSO	9.89	9.45	12 (5/7)	4 (3/1)	122	96.00	96.10	95.90	0.9610	8	171	441
			PSO-GS	9.89	9.49	12 (5/7)	4 (3/1)	121	96.00	96.10	95.90	0.9610	4	107	653
			PSO-DOE	**10**	**9.47**	**12 (5/7)**	**4 (3/1)**	**121**	**96.00**	**96.10**	**95.90**	**0.9610**	**1**	**40**	**509**

На основе данных таблицы 1 можно сделать вывод, что PSO-GS-алгоритм и PSO-DOE-алгоритм решают задачу поиска оптимальных значений параметров SVM-классификатора эффективнее, чем базовый PSO-алгоритм. В частности, сокращается время поиска оптимального решения приблизительно в 3–5 раз, и достигаются лучшие значения показателей качества SVM-классификатора, т. е. наиболее высокие значения показателей общей точности *OSR*, чувствительности *Se* и специфичности *Sp*, а также меньшие значения количества опорных векторов.

При этом именно использование PSO-DOE-алгоритма в большинстве случаев обеспечивает лучшую скорость сходимости к оптимальному решению (т. е. меньшее время первого обнаружения оптимального решения).

5. Заключение

Результаты экспериментальных исследований подтверждают целесообразность использования предлагаемых гибридных версий PSO-алгоритма в рамках решения задачи построения эффективного SVM-классификатора. Достоинство гибридизации базового PSO-алгоритма с алгоритмами поиска по сетке заключается в сокращении

временных затрат на поиск оптимальных значений параметров SVM-классификатора при сохранении, а в некоторых случаях — и улучшении качества классификационных решений.

Полученные результаты были достигнуты благодаря объединению возможностей PSO-алгоритма с положительными чертами алгоритмов поиска по сетке. В частности, был реализован дополнительный поиск по сетке в области потенциального глобально «лучшего» положения частиц в рое с целью дополнительного обновления популяции роя частиц и удаления «худших» частиц.

Дальнейшие исследования могут быть связаны с разработкой рекомендаций по применению гибридных оптимизационных алгоритмов в рамках решения задачи построения SVM-классификаторов для несбалансированных наборов данных.

Литература

[1] *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.

[2] *Клюева И. А.* Гибридный алгоритм настройки параметров интеллектуального классификатора данных // *Математические методы в технике и технологиях*. 2015. Т. 7. С. 234–238.

[3] *Demidova L., Sokolova Yu., Klyueva I., Stepanov N., Tyart N.* Intellectual Approaches to Improvement of the Classification Decisions Quality On the Base Of the SVM Classifier // XII International Symposium «Intelligent Systems-2016» (INTELS'2016). 2016. P. 156–161.

[4] *Joachims T.* A support vector method for multivariate performance measures // In Proceedings of the International Conference on Machine Learning (ICML). — 2005. P. 377–384.

[5] *Yu L., Wang S., Lai K. K., Zhou L.* Bio-Inspired Credit Risk Analysis. Computational Intelligence with Support Vector Machines — Berlin Heidelberg : Springer-Verlag, 2008.

[6] *Vapnik V.* Statistical Learning Theory. — New York : John Wiley & Sons, 1998.

[7] *Ren Y., Bai G.* Determination of optimal SVM parameters by using GA/PSO // *Journal of Computers*. 2010. Vol. 5. No. 8. P. 1160–1168.

[8] *Демидова Л. А., Клюева И. А.* Разработка и исследование гибридных версий алгоритма роя частиц на основе алгоритмов поиска по сетке // *Вестник Рязанского государственного радиотехнического университета*. 2016. № 3 (57). С. 107–117.

[9] *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.

[10] *Карпенко А. П.* Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой. — М. : Изд-во МГТУ им. Н. Э. Баумана, 2014.

[11] *Клюева И. А.* Исследование характеристик сходимости алгоритма роя частиц и его модификации в решении задачи глобальной оптимизации // *Современные технологии в науке и образовании*. 2016. Т. 2. С. 46–50.

[12] *Клюева И. А*. Подходы к модификации алгоритма роя частиц // *Информационные технологии в процессе подготовки современного специалиста*. 2015. Вып. 19. С. 40–46.

[13] *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.

[14] *Demidova L., Klyueva I., Pylkin A.* The Study of Characteristics of the Hybrid Particle Swarm Algorithm in Solution of the Global Optimization Problem // 2016 5th Mediterranean Conference on Embedded Computing (MECO). — IEEE, 2016. P. 322–325.

[15] *Sun J., Lai C. H., Wu X. J.* Particle Swarm Optimisation: Classical and Quantum Perspectives. CRC Press, 2011.

[16] *Hu X., Eberhart R. C., Shi Y.* Particle swarm with extended memory for multiobjective optimization // 2003 IEEE Swarm Intelligence Symposium Proceedings. — IEEE Service Center : Indianapolis, 2003. P. 193–197.

Авторы:

Лилия Анатольевна Демидова — доктор технических наук, профессор кафедры вычислительной и прикладной математики Рязанского государственного радиотехнического университета

Ирина Алексеевна Клюева — аспирант кафедры вычислительной и прикладной математики Рязанского государственного радиотехнического университета

Development of the SVM Classifier by means of the Hybrid Versions of the Particle Swarm Optimization Algorithm Based on the Grid Search

Liliya Demidova, Irina Klyueva

Ryazan State Radio Engineering University
Gagarin Str., 59/1, Ryazan, Russian Federation, 390005

e-mail: liliya.demidova@rambler.ru, i.aleschenko@yandex.ru

Abstract. The approaches to the problem solving of searching of the parameters of the SVM classifier based on the hybridization of the particle swarm optimization algorithm (PSO algorithm) and the grid search algorithms with the aim of providing of high quality classification decisions have been considered. The paper presents two hybrid versions of the basic PSO algorithm, involving the use of the classical Grid Search (GS) algorithm and Design of Experiment (DOE) algorithm correspondingly. It is proposed to use the canonical PSO-algorithm as the basic algorithm. The results of experimental studies confirm the application efficiency of the hybrid versions of the basic PSO-algorithm with the aim of reducing of the time expenditures for searching the optimum parameters of the SVM classifier while maintaining of high quality of its classification decisions.

Keywords: classification, classification quality indicator, hybridization, particle swarm optimization algorithm, grid search algorithm, SVM classifier, regularization parameter, radial basis kernel function.

References

[1] *Demidova L. A., Sokolova Y. S.* (2015) *Vestnik Rjazanskogo gosudarstvennogo radiotehnicheskogo universiteta*, 53:84–92. [In Rus]

[2] *Klyueva I. A.* (2015) Gibridnyj algoritm nastrojki parametrov intellektual'nogo klassifikatora dannyh. In Book Matematicheskie metody v tehnike i tehnologijah, vol. 7, pp. 234–238. [In Rus]

[3] *Demidova L., Sokolova Yu., Klyueva I., Stepanov N., Tyart N.* (2016) Intellectual Approaches to Improvement of the Classification Decisions Quality On the Base Of the SVM Classifier. In Proc. XII International Symposium "Intelligent Systems-2016" (INTELS'2016), pp. 156–161.

[4] *Joachims T.* (2005) A support vector method for multivariate performance measures. In Proc. of the International Conference on Machine Learning (ICML), pp. 377–384.

[5] *Yu L., Wang S., Lai K. K., Zhou L.* (2008) Bio-Inspired Credit Risk Analysis. Computational Intelligence with Support Vector Machines. Berlin Heidelberg, Springer-Verlag.

[6] *Vapnik V.* (1998) Statistical Learning Theory. New York, John Wiley & Sons.

[7] *Ren Y., Bai G.* (2010) *Journal of Computers*, **5**(8):1160–1168.

[8] *Demidova L. A., Klyueva I. A.* (2016) *Vestnik Rjazanskogo gosudarstvennogo radiotehnicheskogo universiteta*, 57:107–117. [In Rus]

[9] *Demidova L. A., Nikulchev E. V., Sokolova Yu. S.* (2016) *Cloud of Science*, **3**(1):5–42. [In Rus]

[10] *Karpenko A. P.* (2014) Sovremennye algoritmy poiskovoj optimizacii. Algoritmy, vdohnovlennye prirodoj. Moscow, MGTU im. N. E. Baumana. [In Rus]

[11] *Klyueva I. A.* (2016) *Sovremennye tehnologii v nauke i obrazovanii*, 2:46–50. [In Rus]

[12] *Klyueva I. A.* (2015) *Informacionnye tehnologii v processe podgotovki sovremennogo specialista*, 19:40–46. [In Rus]

[13] *Kurejchik V. M., Kazharov A. A.* (2011) *Izvestija Juzhnogo federal'nogo universiteta. Tehnicheskie nauki*, 120:30–36. (In Rus)

[14] *Demidova L., Klyueva I., Pylkin A.* (2016) The Study of Characteristics of the Hybrid Particle Swarm Algorithm in Solution of the Global Optimization Problem. In Proc 2016 5th Mediterranean Conference on Embedded Computing (MECO), pp. 322–325.

[15] *Sun J., Lai C. H., Wu X. J.* (2011) Particle Swarm Optimisation: Classical and Quantum Perspectives. CRC Press.

[16] *Hu X., Eberhart R. C., Shi Y.* (2003) Particle swarm with extended memory for multiobjective optimization. In Proc. 2003 IEEE Swarm Intelligence Symposium, pp. 193–197.