

Аспекты реализации генетического алгоритма в задаче смешанной оптимизации в пространстве проектирования переменной размерности

Н.А. Демидов

Аннотация—В статье рассмотрена задача смешанной оптимизации в пространстве проектирования переменной размерности. Цель работы – исследование подхода к решению задачи смешанной оптимизации в пространстве проектирования переменной размерности, обеспечивающего для базового генетического алгоритма возможность переключения между пространствами проектирования разной размерности, на примере задачи настройки параметров SVM-классификатора. Для адаптации базового генетического алгоритма к решению рассматриваемой задачи в хромосому вводится дополнительный ген, кодирующий размерную переменную, которая позволяет переключаться между пространствами проектирования разной размерности в процессе оптимизации, определяя, какие группы генов активны, а какие – скрыты (пассивны), при этом генетический алгоритм поддерживает кодирование в генах хромосомы параметров разного типа с использованием кода Грея. Кодированные параметры могут быть как общими для разных пространств проектирования, так и уникальными (индивидуальными), т.е. сопоставленными только одному конкретному пространству проектирования. Результаты экспериментальных исследований, полученные на примере задачи настройки параметров SVM-классификатора, подтверждают целесообразность выполнения адаптации базового генетического алгоритма. Такой генетический алгоритм позволяет переключаться между пространствами проектирования разной размерности в процессе смешанной оптимизации, обеспечивая одновременный поиск значений оптимизируемых параметров в пространствах разной размерности и получение высоких значений метрик качества классификации, в частности, получение высокого значения метрики F_1 -score.

Ключевые слова— задача смешанной оптимизации, пространство проектирования переменной размерности, размерная переменная, генетический алгоритм, SVM-алгоритм, код Грея, набор данных.

I. ВВЕДЕНИЕ

В настоящее время существуют различные метаэвристические алгоритмы, применяемые для решения задачи оптимизации параметров [1]. В основном такие алгоритмы работают либо с дискретными, либо с непрерывными параметрами. Большинство практических задач относятся к задачам смешанной оптимизации и исполь-

зуют параметры различных типов, таких как дискретные, непрерывные и категориальные. В следствие этого многие метаэвристические алгоритмы оптимизации не применимы к решению задач смешанной оптимизации.

Генетический алгоритм является метаэвристическим алгоритмом, основанным на принципах естественного отбора и генетики. Генетический алгоритм находит оптимальное решение через итеративную эволюцию популяции хромосом, которые подвергаются случайному скрещиванию и мутации на каждом шаге алгоритма. В практических реализациях генетического алгоритма оптимизируемые параметры кодируются генами и могут быть представлены как в бинарном формате, так и в вещественном. Набор генов, представляющий одно из возможных решений задачи оптимизации, называется хромосомой. В случае работы с бинарным форматом для кодировки параметров принято использовать код Грея [2], при применении которого последовательные значения отличаются только в одном двоичном разряде, что делает операции скрещивания и мутации в генетическом алгоритме более управляемыми и предсказуемыми.

В задаче смешанной оптимизации могут одновременно использоваться несколько различных метаэвристических алгоритмов, позволяющих работать с разными типами данных [3, 4]. При этом существенный интерес представляют подходы к решению задач смешанной оптимизации с применением одного метаэвристического алгоритма. В таких алгоритмах код Грея может быть использован для одновременной работы как и с дискретными, так и непрерывными типами параметров.

Сложность задачи оптимизации увеличивается при использовании пространства проектирования переменной размерности [5, 6]. В таких пространствах проектирования число оптимизируемых параметров может изменяться. Один из подходов к работе с пространством проектирования переменной размерности предполагает использование размерной переменной, позволяющей переключаться между пространствами проектирования разной размерности [6]. Значение размерной переменной определяет, с пространством какой размерности в текущий момент времени ведется работа, при этом гены, соответствующие параметрам этого пространства, будут активными, а остальные гены будут скрытыми (пассивными).

Статья получена 10 марта 2026.

Н.А. Демидов, МИРЭА – Российский технологический университет, Москва, Россия (e-mail: nick.a.demidov@rambler.ru).

Целью настоящей работы является исследование различных аспектов реализации генетического алгоритма в задачах смешанной оптимизации в пространстве проектирования переменной размерности. При этом для описания пространства переменной размерности будет использована соответствующая размерная переменная. Кроме того, будут рассмотрены различные способы группировки оптимизируемых параметров с использованием уникальных и общих параметров.

II. ПОДХОДЫ К КОДИРОВАНИЮ ХРОМОСОМЫ В ГЕНЕТИЧЕСКОМ АЛГОРИТМЕ

В генетическом алгоритме оптимизируемые параметры кодируются с применением генов. В задаче смешанной оптимизации кодируемые параметры могут быть представлены в виде целых и вещественных чисел, а также в бинарном формате. Кроме того, в задаче смешанной оптимизации могут быть задействованы и категориальные параметры, которые могут быть закодированы (с учётом ряда соглашений) целыми числами. С учетом вышесказанного, в задаче смешанной оптимизации целесообразно для кодирования оптимизируемых параметров разного типа использовать код Грея, предполагая применение одного алгоритма оптимизации – генетического (без разделения задачи оптимизации на подзадачи, в которых целочисленные параметры оптимизируются отдельно от вещественных).

В случае решения задачи смешанной оптимизации в пространстве проектирования переменной размерности предлагается использовать размерную переменную, значения которой связаны с пространствами параметров разной размерности. Эти параметры могут иметь произвольный тип.

При работе с параметрами целого типа (а также – с категориальными параметрами, для представления которых используются целые числа) необходимо контролировать проблему выхода значений соответствующих генов за границы допустимого диапазона (как на этапе кодирования параметров, так и при реализации операций скрещивания и мутации в генетическом алгоритме).



Рис. 1. Визуализация отношений гена, кодирующего размерную переменную, и генов, определяющих параметры в пространстве проектирования

При работе с параметрами вещественного типа необходимо контролировать точность их представления (например, такие параметры должны быть закодированы с точностью $\varepsilon = 10^{-5}$). Размерную переменную принято кодировать целыми числами, поэтому для неё

необходимо контролировать проблему выхода значений соответствующего гена за границы допустимого диапазона.

Так как каждый кодируемый параметр принимает значения из некоторого допустимого диапазона, то каждый ген содержит некоторое число бит. При этом разные параметры кодируются разным числом бит (в зависимости от размера диапазона, на котором он определен).

Следует отметить, что в данной работе используется одна размерная переменная, для кодирования которой используется представление целых чисел в бинарном формате на основе кода Грея. При этом возможно использование нескольких размерных переменных, принимающих только значения «0» и «1». В этом случае каждая размерная переменная отвечает за активацию конкретного набора параметров, соответствующих некоторому пространству проектирования.

В случае работы с одной размерной переменной в зависимости от её значения используются пространства проектирования различной размерности. В зависимости от используемого пространства проектирования гены хромосомы могут быть либо активными, либо скрытыми (пассивными). Можно считать, что пространство проектирования с активными генами, является активным, а пространство проектирования со скрытыми генами – скрытым. Кроме того, один и тот же ген может быть активным при использовании разных пространств проектирования, то есть гены могут быть как уникальными для каждого пространства, так и общими. Следует отметить, что только активные гены участвуют в оценке качества текущей хромосомы (т.е. в оценке качества текущего решения).

На рисунке 1 представлена визуализация взаимодействия гена, определяющего значение размерной переменной, и генов, определяющих параметры в пространствах проектирования.

Так как размерная переменная принимает значения 0, 1 и 2, то задача оптимизации должна решаться в трёх пространствах проектирования, размерности которых равны соответственно 2, 3 и 1 (рисунок 1). При этом все гены в представленной на рисунке 1 хромосоме являются уникальными.

Все гены в хромосоме, в том числе – ген, кодирующий размерную переменную, могут участвовать в операциях скрещивания и мутации. При этом в зависимости от значения размерной переменной активными будут только те гены, которые напрямую связаны с значением размерной переменной.

III. АДАПТАЦИЯ ГЕНЕТИЧЕСКОГО АЛГОРИТМА К РЕШЕНИЮ ЗАДАЧИ СМЕШАННОЙ ОПТИМИЗАЦИИ В ПРОСТРАНСТВЕ ПРОЕКТИРОВАНИЯ ПЕРЕМЕННОЙ РАЗМЕРНОСТИ

В [6] показано, что введение в хромосому дополнительных генов, отвечающих за размерные переменные, не приводит к изменению основных механизмов и операторов генетического алгоритма. В частности, без потери общности можно использовать оператор турнирной селекции, оператор имитации бинарного скрещивания (Simulated Binary Crossover) [7] и оператор полиномиальной мутации (Polynomial Mutation) [8]. Таким образом, для решения задачи смешанной оптимизации в пространстве проектирования переменной размерности может быть использован генетический алгоритм, работающий с хромосомами, кодирование параметров в которых осуществляется на основе кода Грея, и адаптированный для работы с задачами смешанной оптимизации.

В настоящей работе выполнена адаптация алгоритма SelfCSHAGA (Self-Configuring Success History-based Adaptation Genetic Algorithm) [9], являющийся самоконфигурируемым генетическим алгоритмом оптимизации с адаптацией на основе истории успеха, развивающим идеи алгоритма SHAGA [10], и доказавший своё преимущество над рядом других современных метаэвристических алгоритмов. Алгоритм SelfCSHAGA использует в своей работе механизм настройки вероятностей применения генетических операторов. В частности, алгоритм SelfCSHAGA такие операторы селекции [9], как пропорциональная; ранговая; турнирная с размером турнира, равным 3, 5 или 7. Кроме того, алгоритм SelfCSHAGA позволяет работать с такими операторами скрещивания [9], как равномерное скрещивание с числом родителей, равным 2 [10]; одноточечное скрещивание; двухточечное скрещивание; равномерное равновероятное скрещивание с числом родителей, равным 3 или 8; равномерное пропорциональное скрещивание с числом родителей, равным 3 или 8; равномерное ранговое скрещивание с числом родителей, равным 3 или 8; равномерное турнирное скрещивание с числом, равным родителям 4 или 8 (при размере турниров, равном 2). Оператор мутации в алгоритме SelfCSHAGA аналогичен оператору полиномиальной мутации [8], применяемому в алгоритме SHAGA [10]. В целом, алгоритм SelfCSHAGA имеет измененный цикл работы, структурно приближенный к циклу алгоритма дифференциальной эволюции, применяет модифицированную процедуру скрещивания, позволяющую адаптировать интенсивность скрещивания, а также использовать селективное давление на этапе скрещивания и многородительское скрещивание. Кроме того, алгоритм SelfCSHAGA реализует интеграцию механизма динамической адаптации вероятностей операторов скрещивания и мутации на основе истории успеха. Следует отме-

тить, что цикл работы алгоритма SelfCSHAGA реализован так [9], что для каждой хромосомы формируется хромосома-потомок, которая заменяет хромосому-родителя при улучшении значения функции приспособленности; при этом текущая хромосома обязательно участвует в операции скрещивания в качестве хромосомы-родителя, а остальные хромосомы-родители выбираются с помощью оператора селекции.

Адаптация алгоритма SelfCSHAGA в контексте решения задачи смешанной оптимизации в пространстве переменной размерности заключается в явном указании типов параметров, закодированных разными генами с целью правильной их обработки в процессе кодирования хромосом и реализации эволюционного поиска. При этом основное внимание уделяется корректному заданию функции приспособленности, в которой должно выполняться сопоставление значения размерной переменной и соответствующих этому значению параметров в активном пространстве проектирования.

IV. АПРОБАЦИЯ АДАПТИРОВАННОГО ГЕНЕТИЧЕСКОГО АЛГОРИТМА НА ЗАДАЧЕ РАЗРАБОТКИ SVM-КЛАССИФИКАТОРА

Для апробации возможностей адаптированного генетического алгоритма к решению задач смешанной оптимизации в пространстве проектирования переменной размерности была выбрана задача настройки SVM-классификатора, в которой необходимо выбрать значение параметра регуляризации C , а также – функцию ядра и значения её параметров, обеспечивающие высокое качество классификации, оцениваемое с применением некоторой метрики качества, например, с применением метрики F_1 -score:

$$F_1\text{-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (1)$$

где: $\text{Precision} = \frac{TP}{TP+FP}$; $\text{Recall} = \frac{TP}{TP+FN}$; TP – число истинно положительных (True Positive) исходов, FP – число ложно положительных (False Positive) исходов, FN – число ложно отрицательных (False Negative) исходов при решении задачи классификации.

A. Аспекты разработки SVM классификатора

Пусть $D = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ – набор данных, используемый при разработке SVM классификатора, где $\mathbf{x}_i \in X$; $y_i \in Y = \{1, \dots, m\}$; $i = \overline{1, n}$; n – число объектов в наборе данных; m – число классов; X – множество объектов \mathbf{x}_i ($i = \overline{1, n}$), описываемых q признаками; Y – набор меток классов объектов [11].

SVM-классификатор обучают на N объектах, а тестирование выполняют на $n - N$ объектах. При этом для оценки качества SVM-классификатора используется процедура -кратной перекрестной проверки.

Используемый при разработке SVM-классификатора SVM-алгоритм предполагает, что число классов m равно 2, и реализует бинарную классификацию данных, строя гиперплоскость, разделяющую классы [11]. При этом каждому объекту $x_i \in X$ ставится в соответствие метка класса $y_i \in Y = \{-1; +1\}$ ($i = \overline{1, n}$).

При разработке бинарного SVM-классификатора решается задача построения гиперплоскости, разделяющей классы. В соответствии с теоремой Куна–Таккера эта задача сводится к задаче квадратичного программирования, которая содержит только двойственные переменные λ_i ($i = \overline{1, N}$) [11]:

$$\begin{cases} \frac{1}{2} \cdot \sum_{i=1}^N \sum_{j=1}^N \lambda_i \cdot \lambda_j \cdot y_i \cdot y_j \cdot k(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^N \lambda_i \rightarrow \min_{\lambda} \\ \sum_{i=1}^N \lambda_i \cdot y_i = 0, \\ 0 \leq \lambda_i \leq C, i = \overline{1, N}, \end{cases} \quad (2)$$

где C ($C > 0$) – значение параметра регуляризации; $k(\mathbf{x}_i, \mathbf{x}_j)$ – функция ядра.

В качестве функции ядра $k(\mathbf{x}_i, \mathbf{x}_j)$ используют линейную, полиномиальную, радиальную базисную и сигмоидную функцию ядра соответственно формулам (3), (4), (5) и (6):

$$k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \bullet \mathbf{x}_j, \quad (3)$$

$$k(\mathbf{x}_i, \mathbf{x}_j) = (\mu \cdot \mathbf{x}_i \bullet \mathbf{x}_j + \pi)^r, \quad (4)$$

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \cdot (\mathbf{x}_i - \mathbf{x}_j) \bullet (\mathbf{x}_i - \mathbf{x}_j)), \quad (5)$$

$$k(\mathbf{x}_i, \mathbf{x}_j) = th(\alpha \cdot \mathbf{x}_i \bullet \mathbf{x}_j + \beta), \quad (6)$$

где μ , π , r , γ , α и β – параметры функций ядра; th – функция гиперболического тангенса; « \bullet » – знак операции скалярного умножения.

При этом полиномиальная, радиальная базисная и сигмоидная функции ядра обеспечивают переход в пространство большей размерности, чем размерность исходного пространства признаков, с целью обеспечения лучшей отделимости объектов, принадлежащих разным классам, друг от друга.

В результате решения задачи (2) определяют опорные векторы, являющиеся объектами исходного набора данных, расположенными вблизи разделяющей гиперплоскости, и несущими всю информацию о принципах разделения классов. Для опорных векторов значения двойственных переменных λ_i отличны от нуля: ($\lambda_i \neq 0$).

Правило классификации, в соответствии с которым определяется класс принадлежности объекта \mathbf{x} , имеет вид [11]:

$$F(\mathbf{x}) = \text{sign}(\sum_{i=1}^N \lambda_i \cdot y_i \cdot k(\mathbf{x}_i, \mathbf{x}) + b), \quad (7)$$

где $b = \omega \cdot \mathbf{x}_i - y_i$; $\omega = \sum_{i=1}^N \lambda_i \cdot y_i \cdot \mathbf{x}_i$.

Следует отметить, что задача бинарной классификации может быть легко трансформирована в задачу многоклассовой классификации с использованием стратегий «один против всех» (one-against-all) и «один против одного» (one-against-one) [12].

В задаче настройки параметров SVM-классификатора выбор функции ядра приводит к выбору пространства проектирования той или иной размерности, так как, например, линейная функция ядра не требует определения каких-либо параметров, радиальная базисная функция ядра работает с одним параметром γ , сигмоидальная работает с двумя параметрами α и β , а полиномиальная – с тремя параметрами μ , π и r . При этом еще одна дополнительная размерность в пространстве проектирования появляется в связи с

необходимостью подбора значения параметра регуляризации C .

В. Способы кодирования хромосомы при разработке SVM классификатора

Для группировки генов, сопоставляемых той или иной функции ядра, целесообразно использовать размерную переменную, которая определяет, какие гены в хромосоме будут активными, а какие гены – скрытыми (пассивными).

При разработке SVM-классификатора размерная переменная определяет, какую функцию ядра будет использовать SVM-классификатор, и в зависимости от выбранной функции ядра активирует гены, которые кодируют параметры этой функции.

Кроме того, размерная переменная может активировать ген, отвечающий за параметр регуляризации C , если этот параметр назначен индивидуально (уникально) в дополнение к параметрам каждой функции ядра. Следует отметить, что параметр регуляризации C может быть и общим для всех функций ядра: в этом случае он кодируется одним общим геном.

В данной работе рассматриваются три различных способа кодирования хромосомы. Эти способы различаются различной группировкой генов на основе концепции уникальных и общих генов.

В первом способе кодирования уникальными являются как параметры каждой функции ядра, так и параметры регуляризации C , сопоставленные каждой функции ядра. В этом способе каждая хромосома содержит 11 генов. На рисунке 2 представлена схема, визуализирующая отношения гена, кодирующего размерную переменную, функций ядра и остальных генов для первого способа кодирования хромосом.

Во втором способе кодирования общий ген кодирует общий параметр регуляризации C , при этом каждой функции ядра сопоставлены уникальные гены, кодирующие её уникальные параметры, которые не использует ни одна другая функция ядра. В этом способе каждая хромосома содержит 8 генов. На рисунке 3 представлена схема, визуализирующая отношения гена, кодирующего размерную переменную, функций ядра и остальных генов для второго способа кодирования хромосом.

В третьем способе кодирования общий ген кодирует общий параметр регуляризации C , кроме того, общие гены кодируют общие параметры функций ядра, если эти параметры совпадают по названию, используемому для функций ядра в библиотеке scikit-learn [13]. В этом способе каждая хромосома состоит из 5 генов. На рисунке 4 представлена схема, визуализирующая отношения гена, кодирующего размерную переменную, функций ядра и остальных генов для третьего способа кодирования хромосом.

На рис. 2–4 обозначения генов представлены в виде обозначений параметров, принятых в библиотеке scikit-learn для параметров функций ядра (4), (5) и (6) [10]. При этом для названий функций ядра (3), (4), (5) и (6) введены соответственно обозначения *linear*, *poly*, *rbf* и *sigmoid*, используемые в библиотеке scikit-learn для обозначения функций ядра.

Так, параметр *gamma* соответствует параметру μ в формуле полиномиальной функции ядра (4), параметру γ

в формуле радиально базисной функции ядра (5) и параметру α в формуле сигмоидной функции ядра (6).

Параметр $coef0$ соответствует параметру π в формуле полиномиальной функции ядра (4) и параметру β в формуле сигмоидной функции ядра (6).

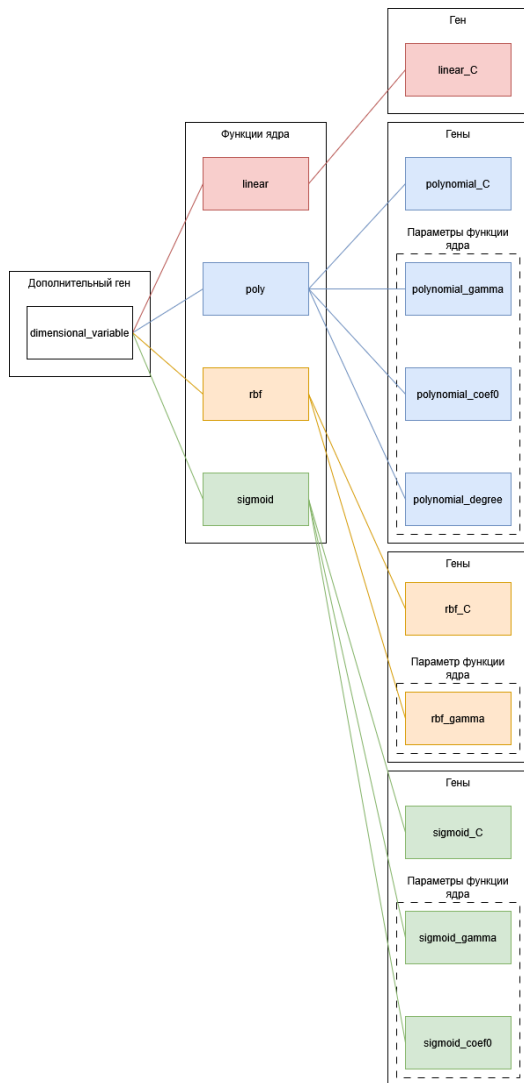


Рис. 2. Визуализация отношений гена, кодирующего размерную переменную, функций ядра и остальных генов в первом способе кодирования

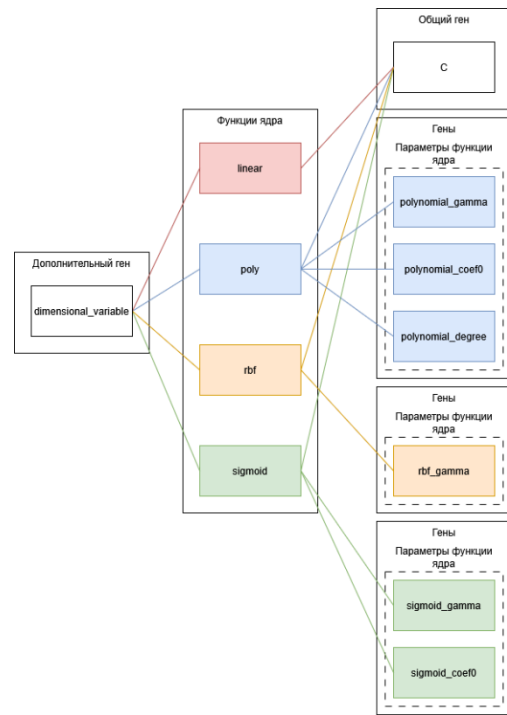


Рис. 3. Визуализация отношений гена, кодирующего размерную переменную, функций ядра и остальных генов во втором способе кодирования

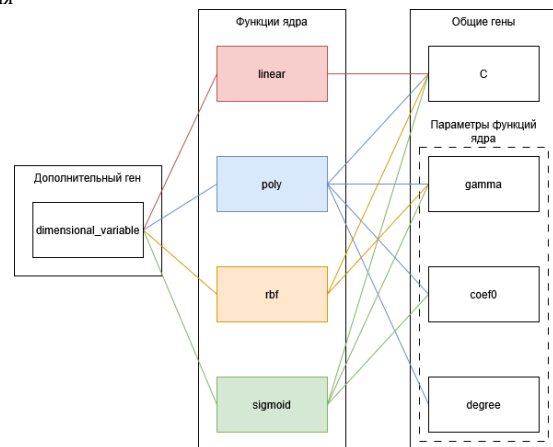


Рис. 4. Визуализация отношений гена, кодирующего размерную переменную, функций ядра и остальных генов для третьего способа кодирования хромосом

Параметр $degree$ соответствует параметру r в формуле полиномиальной функции ядра (4).

Кроме того, на рисунках 2 и 3 к обозначениям генов $gamma$, $coef0$ и $degree$ добавлены префиксы, позволяющие понять, какой функции ядра сопоставлен тот или иной ген (например, ген $polynomial_gamma$ сопоставлен на рисунках 2 и 3 полиномиальной функции ядра, а ген rbf_gamma на этих же рисунках сопоставлен радиальной базисной функции ядра).

Следует отметить, что на всех рис. 2–4 обозначение $dimensional_variable$ соответствует дополнительному гену, определяющему размерную переменную.

V. ЭКСПЕРИМЕНТАЛЬНЫЕ НАБОРЫ ДАННЫХ

В качестве экспериментальных наборов данных рассматриваются три известных набора данных, предлагаемых библиотекой scikit-learn. В таблице 1 приведена информация о используемых наборах данных.

Таблица 1. Характеристики наборов данных

Название набора данных	Число объектов	Число признаков	Число классов
IRIS [14]	150	4	3
WINE [15]	178	13	3
BREAST_CANCER [16]	569	30	2

IRIS – набор данных, признаки в котором представляют физические измерения цветков ириса. Класс в этом наборе данных соответствует виду ириса, к которому принадлежит цветок.

WINE – набор данных, признаки в котором являются результатами химического анализа вин. Класс в этом наборе данных позволяет определить винодельню, в которой было произведено вино.

BREAST_CANCER – набор данных, содержащий признаки полученные из изображений биопсии опухолей молочной железы. Класс в этом наборе данных указывает на то, является ли опухоль доброкачественной или злокачественной.

VI. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Экспериментальные исследования выполнены на языке Python 3.12 в среде Google Colab с применением программной реализации SVM-алгоритма [17], используемого при разработке SVM-классификатора, и алгоритма SelfCSHAGA [18, 19], адаптированного к решению задачи смешанной оптимизации в пространстве проектирования смешанной размерности.

В ходе экспериментальных исследований реализованы три подхода к кодированию хромосом, в результате применения которых хромосомы в трех реализациях генетического алгоритма содержат 11, 8 и 5 генов, кодирующих оптимизируемые параметры.

Длина хромосом в эти трех реализациях в битах равна соответственно 184, 124 и 64 битами.

Согласно рисунку 2 в хромосоме 1-й ген кодирует параметр регуляризации C (для случая использования в SVM-классификаторе линейной функции ядра), 2-й, 3-й, 4-й и 5-й гены кодируют соответственно параметр регуляризации C , а также параметры $poly_gamma$, $poly_coef0$, $poly_degree$ (для случая использования в SVM-классификаторе полиномиальной функции ядра), 6-й и 7-й гены кодируют соответственно параметр регуляризации C , а также параметр rbf_gamma (для случая использования в SVM-классификаторе радиальной базисной функции ядра), 8-й, 9-й и 10-й гены кодируют соответственно параметр регуляризации C , а также параметры $sigmoid_gamma$ и $sigmoid_coef0$ (для случая использования в SVM-классификаторе сигмоидной функции ядра), 11-й ген кодирует размерную переменную, отвечающую за выбор функции ядра. Все гены с 1-го по 10-й в такой хромосоме уникальные, 11-й ген не относится к общим или уникальным. При таком кодировании на каждый ген хромосомы (с 1-го по 11-й) отведенное число бит равно: 20, 20, 20, 20, 2, 20, 20, 20, 20, 20 и 2.

Согласно рис. 3 в хромосоме 1-й ген кодирует параметр регуляризации C , который является общим для всех случаев использования функций ядра, 2-й, 3-й и 4-й

гены кодируют соответственно параметры $poly_gamma$, $poly_coef0$, $poly_degree$ (для случая использования в SVM-классификаторе полиномиальной функции ядра), 5-й ген кодирует параметр rbf_gamma (для случая использования в SVM-классификаторе радиальной базисной функции ядра), 6-й и 7-й гены кодируют соответственно параметры $sigmoid_gamma$ и $sigmoid_coef0$ (для случая использования в SVM-классификаторе сигмоидной функции ядра), 8-й ген кодирует размерную переменную, отвечающую за выбор функции ядра. Гены со 2-го по 7-й, кодирующие параметры разных функций ядра, являются уникальными, 1-й ген является общим (как уже отмечено выше), 8-й ген не относится к общим или уникальным. При таком кодировании на каждый ген хромосомы (с 1-го по 8-й) отведенное число бит равно: 20, 20, 20, 2, 20, 20, 20 и 2.

Согласно рисунку 4 в хромосоме 1-й ген кодирует параметр регуляризации C , который является общим для всех случаев использования в SVM-классификаторе трёх функций ядра, 2-й ген кодирует параметр $gamma$, который является общим для всех случаев использования в SVM-классификаторе трёх функций ядра, 3-й ген кодирует параметр $coef0$, который является общим для случаев использования в SVM-классификаторе полиномиальной и сигмоидной функций ядра, 4-й ген кодирует параметр $degree$, который является уникальным для случая использования в SVM-классификаторе сигмоидной функции ядра, 5-й ген кодирует размерную переменную, отвечающую за выбор функции ядра. При этом 5-й ген не относится к общим или уникальным. При таком кодировании на каждый ген хромосомы (с 1-го по 4-й) отведенное число бит равно: 20, 20, 20, 2, и 2.

Для каждой реализации адаптированного генетического алгоритма выполнено 20 запусков с целью определения для каждой функции ядра, используемой при разработке SVM-классификатора, частоты выбора её в качестве лучшей. При этом популяция содержит 50 хромосом, а число поколений равно 100. При реализации адаптированного генетического алгоритма использовались настройки, заданные по умолчанию для алгоритма SelfCSHAGA. В качестве функции приспособленности используется метрика F_1 -score, значение которой должно быть максимизировано. Метрика F_1 -score используется для оценки качества разработанных SVM-классификаторов. Следует отметить, что при разработке SVM-классификатора использована 5-fold кросс-валидация, а наборы данных подвергнуты масштабированию с использованием стандартизации.

В процессе разработки SVM-классификатора для оптимизируемых параметров SVM-алгоритма определены диапазоны изменения их значений. Так, параметры C , $gamma$ и $coef0$ могут принимать значения из диапазонов, которые определены одинаковым образом как [0.01, 10.0]. Параметр $degree$ может принимать значения из множества {2, 3, 4}, кодируемые в хромосоме числами 1, 2 и 3. Размерная переменная может принимать значения из множества {0, 1, 2, 3}, где значения 0, 1, 2 и 3 соответствуют линейной, полиномиальной, радиальной базисной и сигмоидной функциям ядра.

В табл. 2 и 3 приведены соответственно общие и детальные результаты экспериментальных исследований для набора данных IRIS. В табл. 4 и 5 приведены соответственно общие и детальные результаты экспериментальных исследований для набора данных WINE. В таблицах 6 и 7 приведены соответственно общие и детальные результаты экспериментальных исследований для набора данных BREAST_CANCER. В таблицах 3, 5 и 7 в каждой строке для каждого варианта кодирования хромосомы жирным шрифтом выделены лучшие значения метрики F_1 -score и соответствующие им значения параметров.

В таблицах 3, 5 и 7 не приведены результаты, относящиеся к сигмоидной функции ядра, т.к. ни в одном из запусков ни для одного набора данных эта функция ядра не выбиралась в качестве лучшей.

Кроме того, для каждого набора данных выполнена разработка SVM-классификатора со значениями параметров, заданными по умолчанию, и получены оценки качества классификации на основе метрики F_1 -score (таблицы 3, 5 и 7).

На рисунке 5 представлены графические зависимости для значений функции приспособленности, определяемой на основе метрики F_1 -score, для лучших хромосом в поколениях в 20 запусках адаптированного генетического алгоритма.

Таблица 2. Общие результаты экспериментов с 20 запусками алгоритма оптимизации с разными структурами хромосом для набора данных IRIS

Структура хромосомы	Соотношение побед функций ядра: <i>linear:poly:rbf:sigmoid</i>	Время, ч
Хромосома с 11 параметрами	13:4:3:0	0.62
Хромосома с 8 параметрами	12:6:2:0	0.83
Хромосома с 5 параметрами	14:6:0:0	0.83

Таблица 3. Детальные результаты экспериментов с 20 запусками алгоритма оптимизации с разными структурами хромосом для набора данных IRIS

Структура хромосомы	Значение F_1 -score при значениях параметров по умолчанию	Значения параметров по умолчанию	Лучшее значение F_1 -score для линейной функции ядра	Значения параметров при лучшем значении F_1 -score для линейной функции ядра	Лучшее значение F_1 -score для полиномиальной функции ядра	Значения параметров при лучшем значении F_1 -score для полиномиальной функции ядра	Лучшее значение F_1 -score для радиальной базисной функции ядра	Значения параметров при лучшем значении F_1 -score для радиальной базисной функции ядра
Хромосома с 11 параметрами	0.960	$C=1.0$ функция ядра: радиальная базисная, γ 'scale''	0.987	$C=0.196$	0.980	$C=3.200$, $degree=2$, $\gamma=2.054$, $coef0=0.019$	0.980	$C=1.539$, $\gamma=0.130$
Хромосома с 8 параметрами	0.960	$C=1.0$ функция ядра: радиальная базисная, γ 'scale'	0.987	$C=0.195$	0.987	$C=0.893$, $degree=2$, $\gamma=0.347$, $coef0=0.360$	0.980	$C=8.830$, $\gamma=0.137$
Хромосома с 5 параметрами	0.960	$C=1.0$ функция ядра: радиальная базисная, γ 'scale'	0.987	$C=0.197$	0.987	$C=0.790$, $degree=3$, $\gamma=0.147$, $coef0=0.973$	-	-

*Если γ 'scale', то в качестве используется $\frac{1}{q \cdot \text{var}(X_{\text{train}})}$, где q – число признаков; $\text{var}(X_{\text{train}})$ – дисперсия данных на обучающей выборке X_{train} .

Таблица 4. Общие результаты экспериментов с 20 запусками алгоритма оптимизации с разными структурами хромосом для набора данных WINE

Структура хромосомы	Соотношение побед функций ядра: <i>linear:poly:rbf:sigmoid</i>	Время, ч
Хромосома с 11 параметрами	10:3:7:0	0.85
Хромосома с 8 параметрами	8:6:6:0	1.03
Хромосома с 5 параметрами	12:2:6:0	1.02

Таблица 5. Детальные результаты экспериментов с 20 запусками алгоритма оптимизации с разными структурами хромосом для набора данных WINE

Структура хромосомы	Значение F_1 -score при значениях параметров по умолчанию	Значения параметров по умолчанию	Лучшее значение F_1 -score для линейной функции ядра	Значения параметров при лучшем значении F_1 -score для линейной функции ядра	Лучшее значение F_1 -score для полиномиальной функции ядра	Значения параметров при лучшем значении F_1 -score для полиномиальной функции ядра	Лучшее значение F_1 -score для радиальной базисной функции ядра	Значения параметров при лучшем значении F_1 -score для радиальной базисной функции ядра
Хромосома с 11 параметрами	0.978	$C=1.0$ функция ядра: радиальная базисная, $\gamma='scale'$ *	0.989	$C=0.228$	0.989	$C=0.032$, $degree=2$, $\gamma=0.099$, $coef0=4.728$	0.989	$C=0.819$, $\gamma=0.071$
Хромосома с 8 параметрами	0.978	$C=1.0$ функция ядра: радиальная базисная, $\gamma='scale'$	0.989	$C=0.089$	0.994	$C=0.010$, $degree=2$, $\gamma=0.123$, $coef0=2.042$	0.994	$C=0.981$, $\gamma=0.034$
Хромосома с 5 параметрами	0.978	$C=1.0$ функция ядра: радиальная базисная, $\gamma='scale'$	0.989	$C=0.084$	0.994	$C=0.031$, $degree=2$, $\gamma=0.270$, $coef0=1.121$	0.994	$C=0.981$, $\gamma=0.021$

*Если $\gamma='scale'$, то в качестве используется $\frac{1}{q \cdot \text{var}(X_{\text{train}})}$, где q – число признаков; $\text{var}(X_{\text{train}})$ – дисперсия данных на обучающей выборке X_{train} .

Таблица 6. Общие результаты экспериментов с 20 запусками алгоритма оптимизации с разными структурами хромосом для набора данных BREAST_CANCER

Структура хромосомы	Соотношение побед функций ядра: <i>linear:poly:rbf:sigmoid</i>	Время, ч
Хромосома с 11 параметрами	17:1:2:0	0.88
Хромосома с 8 параметрами	16:3:1:0	1.19
Хромосома с 5 параметрами	16:2:2:0	1.18

Таблица 7. Детальные результаты экспериментов с 20 запусками алгоритма оптимизации с разными структурами хромосом для набора данных BREAST_CANCER

Структура хромосомы	Значение F_1 -score при значениях параметров по умолчанию	Значения параметров по умолчанию	Лучшее значение F_1 -score для линейной функции ядра	Значения параметров при лучшем значении F_1 -score для линейной функции ядра	Лучшее значение F_1 -score для полиномиальной функции ядра	Значения параметров при лучшем значении F_1 -score для полиномиальной функции ядра	Лучшее значение F_1 -score для радиальной базисной функции ядра	Значения параметров при лучшем значении F_1 -score для радиальной базисной функции ядра
Хромосома с 11 параметрами	0.979	$C=1.0$ функция ядра: радиальная базисная, $\gamma='scale'$ *	0.982	$C=0.147$	0.981	$C=0.730$, $degree=2$, $\gamma=0.021$, $coef0=2.786$	0.981	$C=2.800$ $\gamma=0.018$
Хромосома с 8 параметрами	0.979	$C=1.0$ функция ядра: радиальная базисная, $\gamma='scale'$	0.982	$C=0.154$	0.984	$C=1.079$, $degree=4$, $\gamma=0.010$, $coef0=1.095$	0.977	$C=5.402$ $\gamma=0.090$
Хромосома с 5 параметрами	0.979	$C=1.0$ функция ядра: радиальная базисная, $\gamma='scale'$	0.982	$C=0.158$	0.979	$C=0.030$, $degree=2$, $\gamma=0.269$, $coef0=3.929$	0.981	$C=3.151$, $\gamma=0.054$

*Если $\gamma='scale'$, то в качестве используется $\frac{1}{q \cdot var(X_{train})}$, где q – число признаков; $var(X_{train})$ – дисперсия данных на обучающей выборке X_{train} .

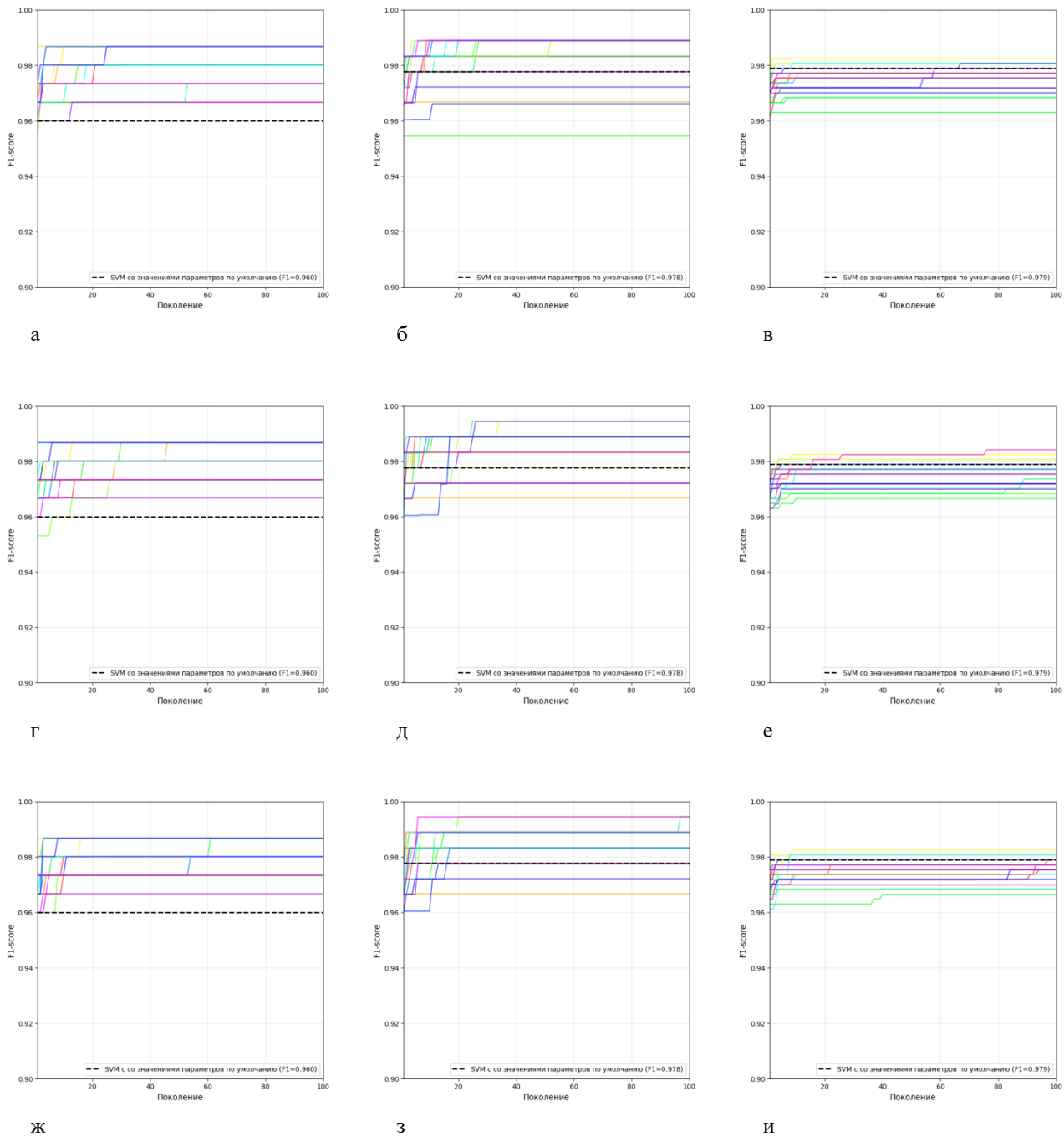


Рис. 5. Визуализация 20 запусков адаптированного генетического алгоритма при кодировании хромосомы разным числом параметров:

- а – запуски с хромосомой с 11 параметрами на наборе данных IRIS;
- б – запуски с хромосомой с 11 параметрами на наборе данных WINE;
- в – запуски с хромосомой с 11 параметрами на наборе данных BREAST_CANCER;
- г – запуски с хромосомой с 8 параметрами на наборе данных IRIS;
- д – запуски с хромосомой с 8 параметрами на наборе данных WINE;
- е – запуски с хромосомой с 8 параметрами на наборе данных BREAST_CANCER;
- ж – запуски с хромосомой с 5 параметрами на наборе данных IRIS;
- з – запуски с хромосомой с 5 параметрами на наборе данных WINE;
- и – запуски с хромосомой с 5 параметрами на наборе данных BREAST_CANCER

Анализ полученных результатов позволяет сделать следующие выводы: применение адаптированного генетического алгоритма позволяет повысить качество SVM-классификаторов в смысле максимизации значения метрики F_1 -score, однако в разных запусках этого алгоритма зачастую наблюдается сходимость к локальным экстремумам, а не к глобальным (таблицы 2, 4 и 6;

рисунок 5). Так, например, для набора данных WINE (таблица 4) для адаптированного генетического алгоритма с хромосомой, кодирующей 11 параметров, в 10 запусках из 20 в качестве лучшей функции ядра выбрана линейная, в 3 запусках из 20 в качестве лучшей функции ядра выбрана полиномиальная, а еще в 7 запусках из 20 в качестве лучшей функции ядра выбрана радиально-

базисная. Можно предположить, что попадание в локальные экстремумы в ряде запусков вызвано введением в структуру хромосомы для переключения между пространствами поиска разной размерности дополнительного гена, кодирующего размерную переменную, и применением в связи с этим концепции активных и скрытых генов. В дальнейшем при решении задачи смешанной оптимизации в пространстве проектирования переменной размерности на основе хромосом с размерной переменной целесообразно исследовать возможности применения принципов нишевых алгоритмов оптимизации [20] с целью повышения надежности поиска решения задачи глобальной оптимизации.

Сравнительный анализ оценок времени (таблицы 2, 4 и 6), затраченного на реализацию 20 запусков адаптированного генетического алгоритма с хромосомами разных длин, не позволяет сделать вывод о том, что с увеличением размерности пространства поиска происходит некоторое увеличение временных затрат на реализацию поиска. В контексте решения задачи настройки параметров SVM-классификатора этот факт может быть объяснен особенностями работы SVM-алгоритма, реализующего поиск опорных векторов и гиперплоскости, разделяющей классы. Эти особенности могут косвенно приводить к увеличению времени оценки значений функции приспособленности на основе метрики F_1 -score отдельных хромосом.

Для набора данных IRIS при всех 3 вариантах кодирования хромосом удалось найти лучшие решения, характеризующиеся одинаковыми значениями метрики F_1 -score, равными 0.987, одинаковыми функциями ядра – линейными, и несколько отличающимися значениями параметра регуляризации C (таблица 3). Кроме того, в 2 вариантах кодирования хромосом (с 8 и 5 генами) удалось найти лучшие решения, характеризующиеся одинаковыми значениями метрики F_1 -score, равными 0.987, одинаковыми функциями ядра – полиномиальными, но отличающимися значениями оптимизируемых параметров (таблица 3).

Для набора данных WINE при 2 из 3 вариантов кодирования хромосом (с 8 и 5 генами) удалось найти лучшие решения, характеризующиеся одинаковыми значениями метрики F_1 -score, равными 0.994, одинаковыми функциями ядра – полиномиальными, но отличающимися значениями оптимизируемых параметров (таблица 5). Кроме того, в этих вариантах кодирования удалось найти лучшие решения, характеризующиеся одинаковыми значениями метрики F_1 -score, равными 0.994, одинаковыми функциями ядра – радиальными базисными, но отличающимися значениями оптимизируемых параметров (таблица 5). При кодировании хромосомы 11 генами лучшее решение характеризуется несколько меньшим значением метрики F_1 -score, равным 0.989 (таблица 5). При этом радиальная базисная функция ядра также выбрана в качестве лучшей (таблица 5).

Для набора данных BREAST_CANCER при варианте кодирования хромосомы с 8 генами удалось найти лучшее решение, характеризующееся значением метрики F_1 -score, равными 0.984, и полиномиальной функцией ядра (таблица 7). При вариантах кодирования хромосом

с 8 и 5 генами удалось найти лучшие решения, характеризующиеся одинаковыми значениями метрики F_1 -score, равными 0.982, одинаковыми функциями ядра – линейными, но и несколько отличающимися значениями параметра регуляризации C (таблица 7).

Анализ графических зависимостей на рисунке 5 показывает, что при использовании в генетическом алгоритме хромосомы с 11 генами, т.е. при использовании только индивидуальных генов, выход на лучшие значения функции приспособленности может осуществляться в более ранних поколениях, чем при использовании в генетическом алгоритме хромосомы с 8 или 5 генами. Аналогичные выводы можно сделать при сравнении графических зависимостей на рисунке 5 для генетических алгоритмов, работающих с хромосомами с 8 или 5 генами: выход на лучшие значения функции приспособленности может осуществляться в более ранних поколениях при использовании в генетическом алгоритме хромосомы с 8 генами, т.е. при преобладании числа индивидуальных генов над числом общих генов.

Следует отметить, что проблема разделения параметров задачи оптимизации на общие и индивидуальные требует более внимательного изучения и исследования в контексте решаемых задач оптимизации, поскольку оперирование значениями параметров, включенными в хромосому в результате применения операторов скрещивания и мутации и оптимизированными под одну функцию ядра, могут не соответствовать специфике другой функции ядра, переход в пространство проектирования которой осуществлен на основе текущего значения размерной переменной.

ЗАКЛЮЧЕНИЕ

Результаты экспериментальных исследований подтверждают, что адаптированный генетический алгоритм позволяет успешно решать задачу смешанной оптимизации в пространстве проектирования переменной размерности, используя значение дополнительного гена, кодирующего в хромосоме размерную переменную, отвечающую за переключение между пространствами проектирования разной размерности в процессе оптимизации. При этом адаптированный генетический алгоритм обеспечивает одновременный поиск значений оптимизируемых параметров разного типа, закодированных генами хромосомы с использованием кода Грея, в пространствах разной размерности и получение высоких значений метрик качества классификации, в частности, получение высокого значения метрики F_1 -score.

В дальнейшем планируется исследовать целесообразность предлагаемого подхода к решению задачи смешанной оптимизации в пространстве переменной размерности на примере задачи настройки параметров одноклассового SVM-классификатора [21], задачи настройки параметров алгоритма PaCMAP [22], а также – задачи настройки параметров современных нейросетевых моделей в сфере текстовой аналитики. Кроме того, планируется исследовать возможности применения концепции нишевых алгоритмов оптимизации [20] с целью повышения надежности поиска решения задачи глобальной оптимизации.

БИБЛИОГРАФИЯ

при реализации алгоритма PaCMAP,” *ИТ-Стандарт*, № 4 (45), 2025, С. 102–123.

- [1] А. П. Карпенко, *Современные алгоритмы поисковой оптимизации: алгоритмы, вдохновленные природой: учебное пособие*. – 2-е изд. – Москва: МГТУ им. Н.Э. Баумана, 2017. – 448 с.
- [2] А. Л. Пережогин, И. С. Быков, “Обзор конструкций и свойств кодов Грея,” *Математические вопросы кибернетики*. Вып. 20, М.: ФИЗМАТЛИТ, 2022, С. 41–60: <https://10.20948/mvk-2022-41>.
- [3] L. Sahoo, A. Banerjee, A. Bhunia, S. Chattopadhyay, “An efficient GA-PSO approach for solving mixed-integer nonlinear programming problem in reliability optimization,” *Swarm and Evolutionary Computation*, vol. 19, 2014, pp. 43–51.
- [4] Y. Gao, Y. Sun, J. Wu, “Difference-genetic co-evolutionary algorithm for nonlinear mixed integer programming problems,” *Journal of Non-linear Science and Its Applications*, vol. 9, no. 3, 2016, pp. 1261–1284.
- [5] J. Pelamatti, L. Brevault, M. Balesdent, E.G. Talbi, Y. Guerin, “How to Deal with Mixed-Variable Optimization Problems: An Overview of Algorithms and Formulations,” in *Advances in Structural and Multi-disciplinary Optimization* // A. Schumacher, T. Vietor, S. Fiebig, K. U. Bletzinger, K. Maute (eds). Springer, Cham, WCSMO 2017. https://doi.org/10.1007/978-3-319-67988-4_5.
- [6] J. Gamot, M. Balesdent, A. Tremolet, R., N.Melab, E.-G. Talbi, “Hidden-variables genetic algorithm for variable Wuilbercq-size design space optimal layout problems with application to aerospace vehicles,” *Engineering Applications of Artificial Intelligence*, vol. 121, 2023, pp. 105941. <https://doi.org/10.1016/j.engappai.2023.105941>.
- [7] K. Deb, R.B. Agrawal, “Simulated binary crossover for continuous search space,” *Complex systems*, vol. 9 (2), 1995, pp. 115–148.
- [8] E.-G. Talbi, *Metaheuristics: from design to implementation*. John Wiley & Sons, 2009, 624 p.
- [9] П. А. Шерстнев, Е. С. Семенкин, “SelfCSHAGA: Самоконфигурируемый генетический алгоритм оптимизации с адаптацией на основе истории успеха,” *Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение*, 2025, № 2 (151), С. 122–139.
- [10] V. Stanovov, S. Akhmedova, E. Semenkin, “Genetic Algorithm with Success History based Parameter Adaptation,” *Proceedings of the 11th International Joint Conference on Computational Intelligence (IJCCI 2019)*. SciTePress, 2019. pp. 180–187. <https://doi.org/10.5220/0008071201800187>.
- [11] O. Chapelle, V. Vapnik, O. Bousquet, S. Mukherjee, “Choosing multiple parameters for support vector machines,” *Machine Learning*, vol. 46, 2002, pp. 131–159. <https://doi.org/10.1023/a:1012450327387>.
- [12] J.-Y. Liu, B.-B. Jia, “Combining One-vs-One Decomposition and Instance-Based Learning for Multi-Class Classification,” *IEEE Access*, vol. 8, 2020, pp. 197499–197507. <https://doi.org/10.1109/ACCESS.2020.3034448>.
- [13] Kernel functions [Электронный ресурс]. – Режим доступа: <https://scikit-learn.org/stable/modules/svm.html#svm-kernels>, свободный (дата обращения: 10.01.2026).
- [14] load_iris [Электронный ресурс]. – Режим доступа: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_iris.html, свободный (дата обращения: 10.01.2026).
- [15] load_wine [Электронный ресурс]. – Режим доступа: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_wine.html, свободный (дата обращения: 10.01.2026).
- [16] load_breast_cancer [Электронный ресурс]. – Режим доступа: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_breast_cancer.html, свободный (дата обращения: 10.01.2026).
- [17] Support vector machines [Электронный ресурс]. – Режим доступа: <https://scikit-learn.org/stable/modules/svm.html#>, свободный (дата обращения: 10.01.2026).
- [18] Thefittest [Электронный ресурс]. – Режим доступа: <https://github.com/sherstpasha/thefittest>, свободный (дата обращения: 10.01.2026).
- [19] P. Sherstnev, “Thefittest: evolutionary machine learning in Python,” *Hybrid Methods of Modeling and Optimization in Complex Systems (HMMOCS-II 2023): Proceedings of the II International Workshop*. Krasnoyarsk: ITM Web of Conferences, vol. 59, 2024, Article 02020, 11 p.
- [20] Y. Matanga, P. Owolawi, C. Du, E. van Wyk, “Nicheing Global Optimisation: Systematic Literature Review,” *Algorithms*, vol. 17(10), 2024, p. 448. <https://doi.org/10.3390/a17100448>.
- [21] Демидов Н.А. Подходы к одноклассовой классификации регулярных выражений // *ИТ-Стандарт*, 2025. – № 2. – С. 32–48.
- [22] Е. Г. Андрианова, Н. А. Демидов, “Исследование сценариев предварительной обработки данных и инициализации вложений

Aspects of Implementing a Genetic Algorithm in Mixed Optimization Problems in A Variable-Size Design Space

N. A. Demidov

Abstract—This paper considers a mixed optimization problem in a variable-size design space. The objective of the study is to explore an approach to solving a mixed optimization problem in a variable-size design space that enables the base genetic algorithm to switch between design spaces of different sizes, using the example of tuning the parameters of an SVM classifier. To adapt the base genetic algorithm to the problem under consideration, an additional gene is introduced into the chromosome. This gene encodes a dimensional variable that enables switching between design spaces of different sizes during the optimization process, determining which gene groups are active and which are hidden (passive). A herewith, the genetic algorithm supports the encoding of parameters of different types in the chromosome genes using the Gray code. The encoded parameters can be either common to different design spaces or unique (individual), i.e., associated only with one specific design space. The experimental results obtained using the example of tuning the parameters of an SVM classifier confirm the feasibility of adapting the base genetic algorithm. This genetic algorithm allows switching between design spaces of different sizes in the process of mixed optimization, ensuring the simultaneous search for values of the optimized parameters in spaces of different sizes and obtaining high values of classification quality metrics, in particular, obtaining a high value of the F_1 -score metric.

Keywords—mixed optimization problem, variable-dimensional design space, dimensional variable, genetic algorithm, SVM algorithm, Gray code, dataset.

REFERENCES

- [1] A. P. Karpenko, *Sovremennyye algoritmy poiskovoy optimizatsii: algoritmy, vdokhnovlennyye prirodoy: uchebnoye posobiye* [Modern Search Engine Optimization Algorithms: Algorithms Inspired by Nature: A Tutorial]. Moscow, 2017, 448 p. (In Russ.).
- [2] A. L. Perezhogin, I. S. Bykov, "Obzor konstruktivnykh svoystv kodov Greya," [Overview of Gray Code Constructions and Properties] *Matematicheskiye voprosy kibernetiki*. Vyp. 20, Moscow: FIZMATLIT, 2022, pp. 41–60. Available: <https://10.20948/mvk-2022-41>.
- [3] L. Sahoo, A. Banerjee, A. Bhunia, S. Chattopadhyay, "An efficient GA-PSO approach for solving mixed-integer nonlinear programming problem in reliability optimization," *Swarm and Evolutionary Computation*, vol. 19, 2014, pp. 43–51.
- [4] Y. Gao, Y. Sun, J. Wu, "Difference-genetic co-evolutionary algorithm for nonlinear mixed integer programming problems," *Journal of Nonlinear Science and Its Applications*, vol. 9, no. 3, 2016, pp. 1261–1284.
- [5] J. Pelamatti, L. Brevault, M. Balesdent, E.G. Talbi, Y. Guerin, "How to Deal with Mixed-Variable Optimization Problems: An Overview of Algorithms and Formulations," in *Advances in Structural and Multidisciplinary Optimization* // A. Schumacher, T. Vietor, S. Fiebig, K. U. Bletzinger, K. Maute (eds). Springer, Cham, WCSMO 2017. https://doi.org/10.1007/978-3-319-67988-4_5.
- [6] J. Gamot, M. Balesdent, A. Tremolet, R., N.Melab, E.-G. Talbi, "Hidden-variables genetic algorithm for variable Wuilbercq-size design space optimal layout problems with application to aerospace vehicles," *Engineering Applications of Artificial Intelligence*, vol. 121, 2023, pp. 105941. <https://doi.org/10.1016/j.engappai.2023.105941>.
- [7] K. Deb, R.B. Agrawal, "Simulated binary crossover for continuous search space," *Complex systems*, vol. 9 (2), 1995, pp. 115–148.
- [8] E.-G. Talbi, *Metaheuristics: from design to implementation*. John Wiley & Sons, 2009, 624 p.
- [9] П.А. Шерстнев, Е.С. Семенкин, "SelfCSHAGA: Самоконфигурируемый генетический алгоритм оптимизации с адаптацией на основе истории успеха," *Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение*, 2025, № 2 (151), С. 122–139.
- [10] V. Stanovov, S. Akhmedova, E. Semenkin, "Genetic Algorithm with Success History based Parameter Adaptation," *Proceedings of the 11th International Joint Conference on Computational Intelligence (IJCCI 2019)*. SciTePress, 2019. pp. 180–187. <https://doi.org/10.5220/0008071201800187>.
- [11] O. Chapelle, V. Vapnik, O. Bousquet, S. Mukherjee, "Choosing multiple parameters for support vector machines," *Machine Learning*, vol. 46, 2002, pp. 131–159. <https://doi.org/10.1023/a:1012450327387>.
- [12] J.-Y. Liu, B.-B. Jia, "Combining One-vs-One Decomposition and Instance-Based Learning for Multi-Class Classification," *IEEE Access*, vol. 8, 2020, pp. 197499–197507. <https://doi.org/10.1109/ACCESS.2020.3034448>.
- [13] Kernel functions [Electronic resource]. – Access mode: <https://scikit-learn.org/stable/modules/svm.html#svm-kernels>, free (date of access: 10.01.2026).
- [14] load_iris [Electronic resource]. – Access mode: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_iris.html, free (date of access: 10.01.2026).
- [15] load_wine [Electronic resource]. – Access mode: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_wine.html, free (date of access: 10.01.2026).
- [16] load_breast_cancer [Electronic resource]. – Access mode: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_breast_cancer.html, free (date of access: 10.01.2026).
- [17] Support vector machines [Electronic resource]. – Access mode: <https://scikit-learn.org/stable/modules/svm.html#>, free (date accessed: 10.01.2026).
- [18] Thefittest [Electronic resource]. – Access mode: <https://github.com/sherstpasha/thefittest>, free (date accessed: 10.01.2026).
- [19] P. Sherstnev, "Thefittest: evolutionary machine learning in Python," *Hybrid Methods of Modeling and Optimization in Complex Systems (HMMOCS-II 2023): Proceedings of the II International Workshop*. Krasnoyarsk: ITM Web of Conferences, vol. 59, 2024, Article 02020, 11 p.
- [20] Y. Matanga, P. Owolawi, C. Du, E. van Wyk, "Nicheing Global Optimisation: Systematic Literature Review," *Algorithms*, vol. 17(10), 2024, p. 448. <https://doi.org/10.3390/a17100448>.
- [21] N. A. Demidov, "Podkhody k odnoklassovoy klassifikatsii regulyarnykh vyrazheniy," [Approaches to one-class classification of regular expressions] *IT-Standart*, № 2, 2025, pp. 32–48.
- [22] E. G. Andrianova, N. A. Demidov, "Issledovaniye stsenariyev predvaritel'noy obrabotki dannykh i initsializatsii vlozheniy pri realizatsii algoritma PaCMAP," [Study of data preprocessing scenarios and initialization of embeddings in the implementation of the PaCMAP algorithm] *IT-Standart*, № 4 (45), 2025, pp. 102–123.

N.A. Demidov, Master's student, MIREA – Russian Technological University, Moscow, Russia (e-mail: nick.a.demidov@rambler.ru).