

Гибридный алгоритм глобальной оптимизации на основе алгоритмов искусственной иммунной системы и роя частиц

03, март 2014

DOI: 10.7463/0314.0700457

профессор, д.ф.-м.н. Карпенко А. П., Щербакова Н. О., Буланов В. А.

УДК 519.6

Россия, МГТУ им. Н.Э. Баумана

apkarpenko@mail.ru

shcherbakova@gmail.com

vb001@rambler.ru

Введение

Для эффективного решения задач глобальной оптимизации в 80-х годах прошлого века началась интенсивная разработка класса стохастических поисковых алгоритмов оптимизации, которые в разных публикациях называют поведенческими, интеллектуальными, метаэвристическими, вдохновленными (инспирированными) природой, роевыми, многоагентными, популяционными и т.д. Используем последний термин. Популяционные алгоритмы (*population algorithms*) предполагают одновременную обработку нескольких вариантов решения задачи оптимизации и представляют собой альтернативу классическим «траекторным» поисковым алгоритмам, в которых в области поиска эволюционирует только один кандидат на решение этой задачи.

Опыт решения сложных задач глобальной оптимизации с помощью популяционных алгоритмов показывает, что применение одного такого алгоритма далеко не всегда приводит к успеху. Поэтому в последние годы большое внимание уделяется гибридизации популяционных алгоритмов глобальной оптимизации. В гибридных алгоритмах, объединяющих различные алгоритмы либо одинаковые алгоритмы, но с различными значениями свободных параметров, эффективность одного алгоритма может компенсировать слабость другого.

Известно большое число методов гибридизации популяционных алгоритмов, так что существует проблема их классификации. Наиболее известные классификации методов гибридизации представлены в работах [1 - 3].

Работа посвящена гибридизации популяционного алгоритма на основе искусственной иммунной системы (*Artificial Immune System, AIS*) и алгоритма роя частиц (*Particle Swarm Optimization, PSO*). В терминах работы [1] используем гибридизацию по схеме препроцессор / постпроцессор (*preprocessor / postprocessor*).

Выделяют следующие основные классы алгоритмов *AIS*: алгоритмы клонального отбора; негативные алгоритмы отбора; иммунные сетевые алгоритмы.

Алгоритмы клонального отбора построены на основе теорий клонального отбора и приобретенного иммунитета. Эти алгоритмы обычно применяют для оптимизации и распознавания образов. Наиболее известным алгоритмом данного класса является *CLONALG* [4]. Модифицированная версия канонического алгоритма *CLONALG*, которую мы используем в данной работе, предложена в 2011 году [5]. Обзор значительного числа алгоритмов этого класса представлен в работе [6].

Негативные алгоритмы отбора основаны на моделировании процессов позитивной и негативной селекции [7]. Этот класс алгоритмов, как правило, используют для классификации и распознавания. *Иммунные сетевые алгоритмы* основаны на теории идиотипических сетей и используются при решении, прежде всего, задач кластеризации и визуализации данных. Известны также варианты этих алгоритмов, предназначенные для решения задач оптимизации [8].

В основу метода роя частиц (*Particle Swarm Optimization, PSO*) положена социально-психологическая поведенческая модель толпы. Известно большое число модификаций этого алгоритма [9].

AIS и *PSO* алгоритмы оптимизации обладают взаимодополняющими свойствами. Так, известно, что алгоритмы *AIS* имеют высокую вероятность локализации глобального экстремума целевой функции, но невысокую скорость сходимости. Напротив, для алгоритмов *PSO* характерна высокая скорость сходимости, но, если не прикладывать дополнительных усилий, относительно невысокая вероятность локализации глобального экстремума. Гибридизация алгоритмов *AIS* и *PSO* преследует цель объединить указанные достоинства этих алгоритмов и нивелировать их недостатки.

В работе предложен гибридный алгоритм, названный *AIS_PSO*, выполнена его программная реализация и исследование эффективности на ряде тестовых функций. С помощью разработанного

алгоритмического и программного обеспечения решена задача оптимального управления вибророботом.

Достоинством вибророботов является то, что они не нуждаются во внешних движителях, и их можно конструктивно выполнять в виде запаянных капсул. Это особенно актуально для медицинских роботов, предназначенных, например, для передвижения по кровеносным сосудам. Исследованию вибророботов посвящено большое число работ. Укажем в качестве примера работу [10], содержащую большой библиографический обзор. В работе [11] представлена математическая модель вибрационного робота, который перемещается по горизонтальной плоскости за счет гармонического движения внутренних тел. Данная модель использована и в нашей работе.

В первом разделе работы дана постановка задачи и представлены базовые алгоритмы ее решения. Во втором разделе рассмотрен предложенный авторами гибридный алгоритм *AIS_PSO*, а также его программная реализация. Третий раздел посвящен исследованию эффективности разработанного алгоритмического и программного обеспечения. В четвертом разделе поставлена задача оптимального управления вибророботом и приведено ее решение, основанное на сведении задачи к задаче нелинейного программирования. В заключении сформулированы основные результаты работы и перспективы ее развития.

1. Постановка задачи и базовые алгоритмы ее решения

Рассматриваем детерминированную непрерывную задачу глобальной безусловной минимизации

$$\min_{X \in R^{|X|}} f(X) = f(X^*) = f^*, \quad (1)$$

где $f(X) \in R^1$ – скалярная целевая функция (критерий оптимальности), $f(X^*) = f^*$ – искомое экстремальное значение целевой функции, $X = (x_1, x_2, \dots, x_{|X|}) \in R^{|X|}$ – $|X|$ -мерный вектор варьируемых параметров; $R^{|X|}$ – $|X|$ -мерное арифметическое пространство.

Алгоритм CLONALG разработан на основе теории клонального отбора иммунной системы [4]. Первоначально алгоритм применялся для решения задач распознавания образов, но затем он был адаптирован для решения задач оптимизации.

Определим основные термины, которые использует алгоритм *CLONALG*.

Антитело (клетка) – кандидат в решение задачи (1), вектор вещественных чисел X .

Аффинность клетки - величина, обозначающая полезность клетки и равная соответствующему значению целевой функции.

Клон клетки - клетка, образованная из данного антитела путем его полного копирования.

Мутация клетки - случайное изменение компонентов вектора решения, соответствующего данному антителу.

Алгоритм *CLONALG* использует следующие принципы теории клонального отбора:

- отбор,
- клонирование лучших антител,
- мутация клонов,
- создание на основе лучших антител множества клеток памяти,
- обеспечение и сохранение разнообразия популяции антител.

Согласно работе [4], схема алгоритма *CLONALG* имеет следующий вид.

1) Создаем случайным образом множество антител $S = \{s_i, i \in [1:|S|]\}$, состоящее из подмножеств M и P , так что $S = M \cup P$. Здесь M – множество клеток *памяти*, P – множество антител, используемых для увеличения разнообразия популяции (см. ниже).

2) Выбираем из множества P лучшие антитела числом n на основании их аффинности, то есть формируем множество антител P^b .

3) Осуществляем клонирование антител из множества P^b и помещаем результаты клонирования во множество C . Число клонов n_C определяем по формуле

$$n_C = \sum_{i=1}^n \left\lceil \frac{b|S|}{i} \right\rceil, \quad (2)$$

где $b > 0$ – свободный параметр, определяющий число клонов каждого из антител; $\lceil \cdot \rceil$ - символ ближайшего целого.

4) Подвергаем множество клонов C мутации, со степенью мутации, обратно пропорциональной их аффинности. Помещаем результаты мутации во множество C^* .

5) Выбираем из множества C^* лучшие антитела и помещаем их во множество клеток памяти M , а также заменяем этими антителами их антитела-родители из множества P , если аффинность антитела-потомка выше аффинности антитела-родителя.

6) Для сохранения разнообразия популяции P заменяем в ней n_d худших антител новыми, сгенерированными случайным образом.

Модифицированный алгоритм CLONALG. В рассмотренном каноническом алгоритме *CLONALG* используется мутация, степень которой обратно пропорциональна аффинности мутирующего антитела. В работе [5] предложена модификация этого алгоритма, предполагающая с вероятностью 50% использование одного из двух операторов мутации.

Схема модифицированного алгоритма *CLONALG* отличается от схемы базового алгоритма только правилами выполнения шага 4, который в данном случае имеет следующий вид [5].

Подвергаем множество клонов C мутации и формируем множество C^* . Антитело s_i подвергается мутации с вероятностью $\frac{\xi_i}{t}$, где

$$\xi_i = \frac{f(X_i)}{\sum_{j=1}^n f(X_j)};$$

t - текущий номер итерации. При этом с вероятностью 50% используем один из двух следующих операторов мутации:

$$X' = X + \frac{u(0;1) U_{|X|}(a^-; a^+) t}{n_C};$$

$$X' = X + \frac{u(0;1) U_{|X|}(a^-; a^+)}{t n_C}.$$

Здесь и далее $u(a^-; a^+)$ - псевдослучайное число, равномерно распределенное в интервале $[a^-, a^+]$; $U_{|X|}(a^-; a^+)$ - $|X|$ -мерный вектор псевдослучайных чисел, все компоненты которого равномерно распределены в том же интервале.

Заметим, во-первых, что вероятность мутации в модифицированном алгоритме *CLONALG*, определяется аффинностью антитела, так что клон, полученный от лучшего антитела, имеют наименьшую вероятность мутации, а от худшего – наибольшую (напомним, что мы рассматриваем задачу минимизации целевой функции). Во-вторых, вероятность мутации уменьшается с ростом номера итерации.

В работе [5] показано, что модифицированный алгоритм *CLONALG* имеет более высокую скорость сходимости, чем базовый алгоритм.

Метод роя частиц [9]. Множество частиц в рое обозначаем $S = \{s_i, i \in [1:|S|]\}$. На итерации t координаты частицы s_i определяют $|X|$ -мерные векторы ее координат $X_{i,t}$ и

«скорости» $V_{i,t}$. Начальные координаты и скорости всех частиц роя полагаем заданными и равными $X_{i,0} = X_i^0$, $V_{i,0} = V_i^0$ соответственно.

Основные итерационные формулы алгоритма *PSO* имеют вид

$$X_{i,t+1} = X_{i,t} + V_{i,t+1},$$

$$V_{i,t+1} = \alpha V_{i,t} + U[0; \beta] \otimes (X_{i,t}^b - X_{i,t}) + U[0; \gamma] \otimes (X_{g,t} - X_{i,t}).$$

Здесь $\otimes$ - символ покомпонентного умножения векторов; $X_{i,t}^b$ - вектор координат частицы S_i с наилучшим (в смысле формулы (1)) значением целевой функции $f(X)$ за все время поиска $[0:t]$; $X_{g,t}$ - вектор координат соседней с данной частицы с наилучшим за то же время значением целевой функции; α , β , γ - свободные параметры алгоритма. Параметр α определяет «инерционные» свойства частицы, β – ее «когнитивные» свойства, γ – «социальность» частицы. Рекомендуемые значения параметров α , β , γ равны, соответственно, 0,7298; 1,49618; 1,49618.

Соседство частиц в рое определяется *топологией соседства частиц*, задаваемой неориентированным графом, в котором вершины графа соответствуют частицам роя, а ребра связывают непосредственных соседей. В алгоритме *PSO* обычно используют топологии «клика» (глобально оптимальная топология), «кольцо» (локально оптимальная топология), «двумерный тор» (топология фон Неймана), «кластер».

Обзор большого числа модификаций алгоритма *PSO* представлен в работе [9].

Гибридный алгоритм *HAIS* предложен в работе [12]. Алгоритм сочетает в себе лучшие свойства базовых алгоритмов - быстрая сходимость алгоритма *PSO* и высокая вероятность локализации глобального экстремума алгоритма клонального отбора.

Схема гибридного алгоритма имеет следующий вид.

- 1) Инициализируем популяцию, состоящую из S частиц, и выполняем фиксированное число итераций алгоритма *PSO*. Выбираем $|S_1| = \frac{|S|}{2}$ лучших полученных частиц.
- 2) Случайным образом создаем $|S_2| = |S_1|$ решений (антител) - формируем популяцию S_2

- 3) Объединяем отобранные частицы и антитела в единую популяцию антител S размером $|S| = |S_1| + |S_2|$ и вычисляем значения целевой функции для каждого из антител.
- 4) Клонировем n лучших антител пропорционально их аффинности – создаем временное множество клонов C .
- 5) Применяем мутацию к множеству C , образуя таким образом множество мутированных клонов C^* .
- 6) Выбираем лучшие элементы из множества C^* для формирования множества клеток памяти M .
- 7) Обновляем множество антител S , записав туда вместо худших антител все клетки памяти и сгенерированные для обеспечения разнообразия популяции n_d антител.

2. Гибридный алгоритм AIS_PSO и его программная реализация

Алгоритм AIS_PSO представляет собой модификацию алгоритма CLONALG + PSO, заключающуюся в замене алгоритма CLONALG этим же модифицированным алгоритмом из работы [5]. Число итераций t_{PSO} алгоритма PSO задает пользователь, а число итераций модифицированного алгоритма CLONALG определяется условием стагнации итерационного процесса в течение заданных пользователем t_{Δ} итераций, когда лучшее достигнутое популяцией значение целевой функции не удастся уменьшить на величину, меньшую или равную ε .

Схема алгоритма AIS_PSO имеет следующий вид.

- 1) Инициализируем рой $S_1(0)$ из $|S_1|$ частиц и популяцию $S_2(0)$ из $|S_2|$ антител.
- 2) Выполняем t_{PSO} итераций алгоритма PSO и получаем таким образом рой $S_1(t_{PSO})$.
- 3) Объединяем популяции $S_1(t_{PSO})$, $S_2(0)$ в единую популяцию S численностью $|S| = |S_1| + |S_2|$.
- 4) Выполняем итерации алгоритма AIS до выполнения условия окончания итераций (условия стагнации вычислительного процесса).

Программа AIS_PSO реализована в среде программного комплекса MatLab (Matrix Laboratory) [13]. Язык программирования MatLab представляет собой высокоуровневый [интерпретируемый язык программирования](http://technomag.bmstu.ru/doc/700457.html), ориентированный на векторные и матричные вычисления. Язык включает в себя широкий спектр функций, в том числе, графических, интегрированную среду разработки, объектно-ориентированные возможности и интерфейсы к программам, написанным на других языках программирования.

Для повышения вероятности локализации глобального экстремума программа AIS_PSO предусматриваем возможность мультистарта, по результатам которого производится статистическая обработка.

3. Вычислительный эксперимент

Исследование эффективности разработанного алгоритмического и программного обеспечения выполнено на следующих хорошо известных тестовых функциях.

1) Задача минимизации одноэкстремальной овражной функции Розенброка (*Rosenbrock*) в гиперкубе

$$D = \{X \mid -b \leq x_i \leq b, i \in [1:|X|]\}, \quad (3)$$

где $b = 30$.

2) Аналогичная задача для четырехэкстремальной функции Химмельблау (*Himmelblau*) и гиперкуба (3), в котором $b = 100$.

3) Такая же задача для многоэкстремальной функции Растригина (*Rastrigin*) и $b = 5,12$.

Во всех задачах размерность вектора варьируемых параметров $|X|$ принята равной 10. В качестве критериев эффективности алгоритма и его программной реализации используем

а) оценку вероятности локализации глобального минимума $\bar{p}$ с заданной точностью $\varepsilon = 10^{-4}$, полученную по 50 стартам программы,

б) среднее число испытаний (вычислений значений целевой функции) $\bar{n}_f$, полученное тем же методом.

Вычислительный эксперимент выполнен при следующих значениях свободных параметров алгоритма AIS_PSO:

- стандартные значения параметров α, β, γ алгоритма PSO, равные 0,7298, 1,49618, 1,49618 соответственно;

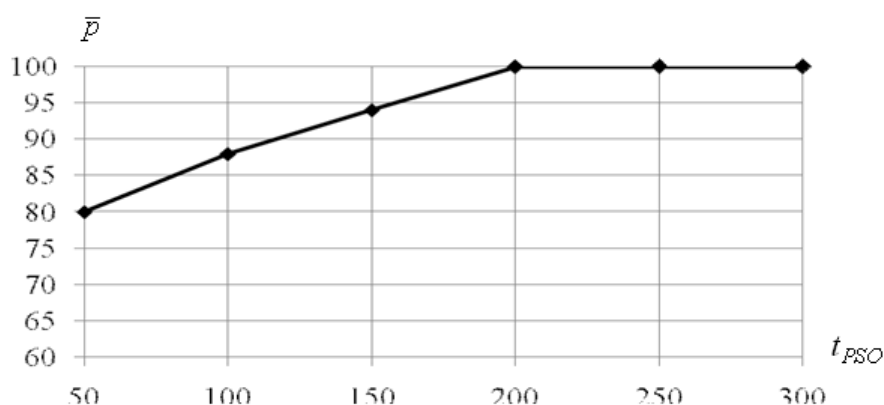
- число антител, подвергающихся клонированию $n = 60$;

– параметр, определяющий число клонов каждого из антител, $b = 0,2$.

Число частиц в алгоритме PSO принято равным $|S_1| = 50$, мощность популяции иммунного алгоритма равной $|S_2| = 300$.

Новым в алгоритме *AIS_PSO*, по сравнению с базовым модифицированным алгоритмом *CLONALG* [5] является свободный параметр t_{PSO} . Поэтому вычислительный эксперимент выполнен при варьировании именно этого параметра.

Оценка вероятности локализации глобального экстремума. Зависимость этой оценки от значений параметра t_{PSO} иллюстрирует рисунок 1. Рисунок показывает, что для функции Растригина оценка вероятности локализации глобального экстремума $\bar{p}$ достигает 100% уже при 200 итерациях алгоритма *PSO*. Для функций Розенброка и Химмельблау получены аналогичные зависимости.



Рису. 1. Оценка вероятности $\bar{p}$ в функции числа итераций t_{PSO} : функция Растригина

Характер зависимости величины $\bar{p}$ от тестовой функции иллюстрирует рисунок 2. Исследование выполнено при $t_{PSO} = 150$.

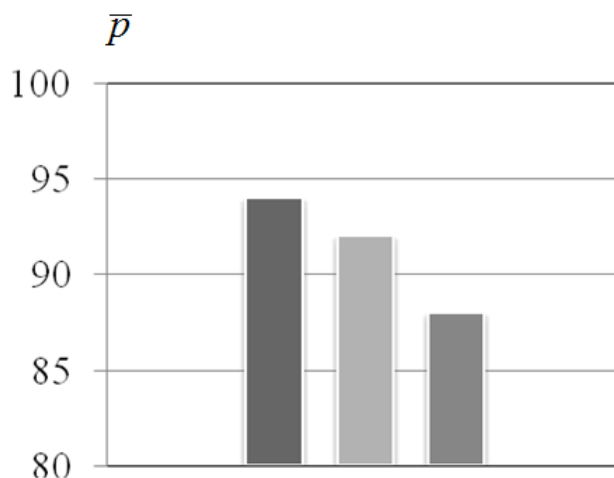


Рис. 2. Оценка вероятности $\bar{p}$ для различных тестовых функций: $t_{PSO} = 150$; ■ – функция Химмельблау, ■ - Розенброка, ■ - Растригина

Результаты данного исследования показывают высокую эффективность алгоритма *AIS_PSO* по критерию вероятности локализации глобального экстремума. Действительно, задача поиска глобального минимума 10-мерной функции Растригина в рассматриваемой области D имеет очень большое число локальных минимумов. Однако даже для такой сложной задачи алгоритм *AIS_PSO* обеспечил 100% вероятность локализации глобального минимума (по результатам 50 стартов).

Среднее число испытаний. На рисунке 3 представлена зависимость среднего числа испытаний $\bar{n}_f$ от величины t_{PSO} для функции Растригина. Для функций Розенброка и Химмельблау получены аналогичные зависимости.

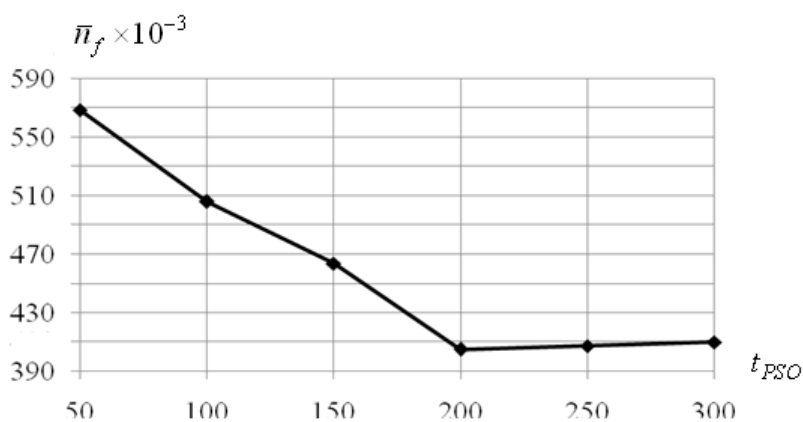


Рис. 3. Среднее число испытаний $\bar{n}_f$ в функции величины t_{PSO} : функция Растригина

Из рисунка 3 ожидаемо следует, что с ростом числа итераций алгоритма *PSO* до примерно 200 среднее число испытаний $\bar{n}_f$ почти линейно уменьшается, а затем медленно увеличивается. Эффект обусловлен тем фактом, что с ростом величины t_{PSO} уменьшается число итераций алгоритма *AIS*. А поскольку популяция этого алгоритма имеет большую мощность, чем популяция алгоритма *PSO*, одновременно уменьшается число вычислений значений целевой функции.

Влияние на величины $\bar{n}_f$ ландшафта целевой функции иллюстрирует рисунок 4, на котором принято $t_{PSO} = 200$.

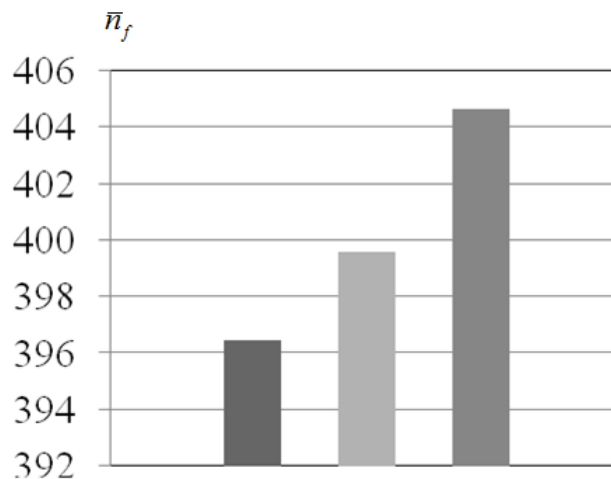


Рис. 4. Среднее число испытаний $\bar{n}_f$ в зависимости от тестовой функции: $t_{PSO} = 200$; ■ – функция Химмельблау, ■ - Розенброка, ■ - Растригина

Представленные результаты не выявили противоречия между быстрой сходимостью алгоритма *PSO* и высокой вероятностью локализации глобального экстремума алгоритма клонального отбора, вследствие которого с ростом величины t_{PSO} следует ожидать уменьшение вероятности локализации глобального экстремума. Есть основания полагать, что эффект проявится при более высокой сложности тестовых функций (более высокая размерность их аргумента и/или более сложный ландшафт).

4. Оптимальное управление вибророботом

Математическая модель виброробота [11]. Рассматриваем виброробот с одним дебалансным ротором. Робот состоит из корпуса, внутри которого на одной горизонтальной оси, расположенной перпендикулярно направлению движения, установлена пара колес, между которыми закреплен груз. Груз закреплен на некотором удалении от оси вращения колес. Таким образом, система, стоящая из пары указанных колес и груза, представляет собой ротор со смещенным центром масс (дебалансный ротор). Угловая скорость вращения ротора может регулироваться.

Полагаем, что между корпусом робота и плоскостью опоры действуют силы сухого и линейного вязкого трения. Указанный груз считаем точечным, массами колес, между которыми он закреплен, пренебрегаем.

Используем инерциальную систему координат Oxy , а также подвижную систему $O'\xi\eta$, жестко связанную с корпусом робота. Оси Ox , $O'\xi$ – горизонтальны, а оси Oy , $O'\eta$ направлены

вертикально вверх. Начала координат O , O' лежат на одной горизонтальной прямой.

Введем еще следующие обозначения: x – координата точки O' в системе координат Oxy (смещение корпуса относительно этой системы); m_1 – масса корпуса робота; m_2 – масса груза; ϖ – скорость вращения ротора; g – ускорение свободного падения.

Во введенных обозначениях и соглашениях математическая модель робота описывается системой обыкновенных дифференциальных уравнений

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = \frac{\Phi_x \sin \varpi t}{M} - \frac{\mu x_2}{M} + \frac{F_c}{M} \end{cases} \quad (4)$$

с нулевыми начальными условиями. Здесь $x_1 = x$, $x_2 = \dot{x}$; $M = m_1 + m_2$; $\Phi_x = m_1 a \varpi^2 \sin \varpi t$ – управляющая сила по горизонтали; μ – коэффициент вязкого трения; F_c – сила сухого трения, определяемая выражением

$$F_c = \begin{cases} -k N \operatorname{sign} x_2, & x_2 \neq 0, \\ -\Phi_x, & x_2 = 0 \text{ и } |\Phi_x| \leq kN, \\ -k N \operatorname{sign} \Phi_x, & x_2 = 0 \text{ и } |\Phi_x| > kN. \end{cases}$$

Здесь приняты обозначения:

$$N = M g + \Phi_y \geq 0;$$

$\Phi_y = m_1 a \varpi^2 \sin (\varpi t + \pi/2)$ – управляющая сила по вертикали; a – радиус окружности, по которой движется центр масс груза.

Необходимо найти закон изменения частоты вращения ротора $\varpi(t)$, который обеспечивает заданный закон движения робота (зависимость перемещения робота от времени) $\varphi(t)$. При всех $t \in [0; T]$ должно выполняться ограничение $\varpi^- \leq \varpi(t) \leq \varpi^+$, где $[0; T]$ – рассматриваемый интервал времени движения робота; ϖ^- , ϖ^+ – заданные константы.

В качестве критерия качества управления используем функционал

$$f(\varpi(t)) = \max_{t \in [0; T]} |x_1(\varpi(t), t) - \varphi(t)|,$$

подлежащий минимизации по допустимому закону $\varpi(t)$:

$$\min_{\varpi(t) \in D_{\varpi}} f(\varpi(t)) = f(\varpi^*(t)). \quad (5)$$

Здесь $D_{\varpi} = \{\varpi(t) \mid \varpi^- \leq \varpi(t) \leq \varpi^+, t \in [0; T]\}$.

Таким образом, имеем задачу оптимального управления динамической системой (4), (5).

Сведение к задаче нелинейного программирования. При всех известных недостатках решения поставленной задачи оптимального управления методом сведения к задаче нелинейного программирования [14], используем именно этот метод.

Покроем интервал $[0; T]$ равномерной сеткой с узлами $t_i, i \in [0; L]$ и будем искать оптимальное управление $\varpi^*(t)$ в классе кусочно-линейных функций (рисунок 5).

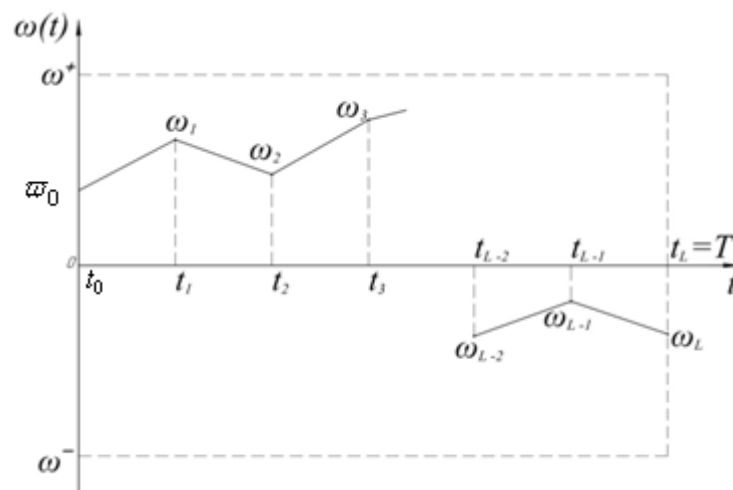


Рис. 5. Кусочно-линейная аппроксимация управления $\varpi(t)$

Положим, что $\varpi_0 = 0$ и обозначим $\Omega = (\varpi_1, \varpi_2, \dots, \varpi_L)$ - L -мерный вектор управления. Таким образом, задача оптимального управления (4), (5) сведена к задаче глобальной условной оптимизации

$$\min_{\Omega \in D_{\Omega}} f(\Omega) = f(\Omega^*), \quad (6)$$

где $D_{\Omega} = \{\varpi_i \mid \varpi^- \leq \varpi_i \leq \varpi^+, i \in [1:L]\}$. Подчеркнем, что для каждого допустимого вектора Ω вычисление значения целевой функции $f(\Omega)$ требует интегрирования системы уравнений (4).

Вычислительный эксперимент выполнен для робота, имеющего следующие параметры: расстояние от оси вращения ротора до центра масс груза $a = 2,3$ см; масса корпуса робота $m_1 = 567$ г; масса груза в роторе $m_2 = 44,5$; коэффициент трения между корпусом робота и опорной плоскостью $\mu = 0,37$. Поскольку системы (4) не является жесткой, для ее интегрирования используем метод Рунге-Кутты четвертого порядка.

Характер движения робота с указанными параметрами иллюстрирует рисунок 6, где принято $V = \dot{x} = x_2$. Рисунок получен при постоянной скорости вращения ротора $\omega(t) = 65,97$ рад/с.

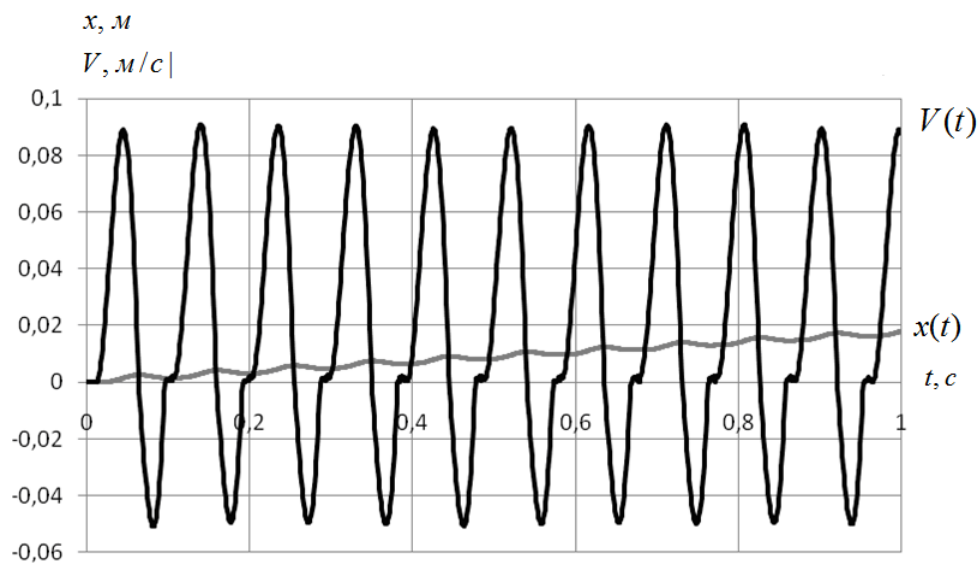


Рис. 6. Характер движения виброробота: $\omega(t) = \text{const} = 65,97$ рад/с

Решение задачи (6) выполнено с использованием разработанного алгоритмического и программного обеспечения *AIS_PSO*. Размерность вектора Ω принята равной 30. Задача решена при следующих значениях свободных параметров алгоритма *AIS_PSO*: $|S| = 300$; $|S_1| = 50$; $|S_2| = 250$; $n = 60$; $b = 0,2$; $t_{PSO} = 200$; $|X| = L = 36$. Результаты решения задачи иллюстрирует рисунок 7.

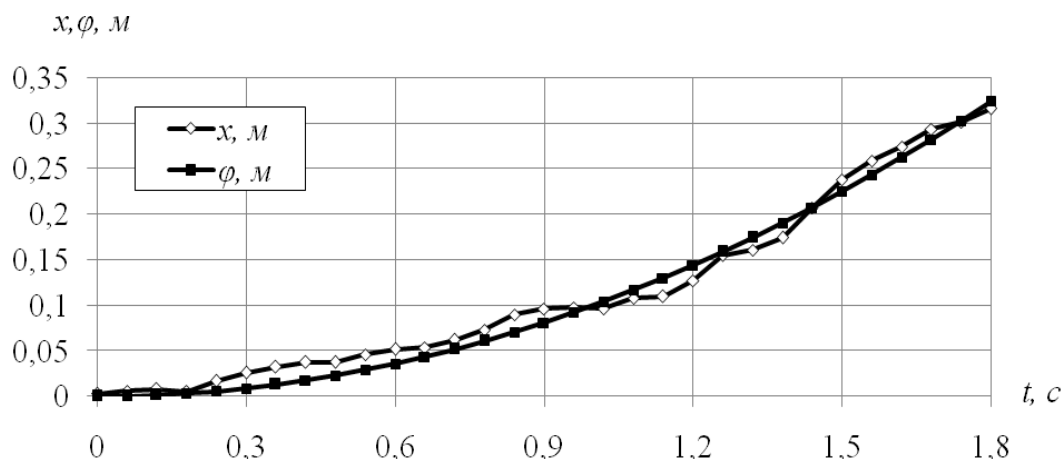


Рис. 7. Заданный $\varphi(t)$ и полученный $x^*(t)$ законы движения виброробота

Результаты решения задачи об оптимальном управлении вибророботом показывают высокую эффективность разработанного алгоритмического и программного обеспечения при решении задач глобальной оптимизации высокой размерности (в данном случае, равной 36) и вычислительной сложности целевой функции.

Заключение

В работе предложен алгоритм *AIS_PSO*, который представляет собой модификацию гибридного алгоритма, построенного на основе иммунного алгоритма *CLONALG* и алгоритма роя части. Модификация заключается в замене алгоритма *CLONALG* этим же, но модифицированным алгоритмом. Средствами программного комплекса *MatLab* выполнена программная реализация предложенного алгоритма. Исследование эффективности разработанного алгоритмического и программного обеспечения выполнено на тестовых функциях Резенброка, Химмельблау и Растригина. Результаты исследования показали эффективность предложенных алгоритмических и программных решений.

С помощью программы *AIS_PSO* решена задача оптимального управления движением вибрационного робота. Использован известный метод решения задачи оптимального управления динамической системой, основанный на сведении этой задачи к задаче нелинейного программирования высокой размерности. Результаты решения задачи также показали высокую эффективность принятых алгоритмических и программных решений.

Результаты исследования породили некоторые вопросы. Так эксперименты с использованием тестовых функций не выявили ожидаемых ускорения сходимости алгоритма *AIS_PSO* и высокой вероятностью локализации глобального экстремума. Дополнительные исследования необходимы также для получения сравнительных многокритериальных оценок эффективности алгоритма *AIS_PSO* с эффективностью его базовых алгоритмов *CLONALG* и роя частиц.

Список литературы

1. Wang X. Hybrid nature-inspired computation method for optimization. Doctoral Dissertation. Helsinki University of Technology, TKK Dissertations, 2009, 161 p.
2. El-Abd, Kamel M. A taxonomy of cooperative search algorithms // In: Hybrid Metaheuristics. Second International Workshop. Vol. 3636. 2005. P. 32-41.
3. Raidl G.R. A Unified View on Hybrid Metaheuristics // In: Hybrid Metaheuristics. Springer Berlin Heidelberg, 2006. P. 1-12. (Ser. Lecture Notes in Computer Science; vol. 4030). DOI: 10.1007/11890584_1
4. Castro L.N., Zuben F.J. Learning and optimization using the clonal selection principle // IEEE Transactions on Evolutionary Computation. 2002. Vol. 6, no. 3. P. 239-251. DOI: [10.1109/TEVC.2002.1011539](https://doi.org/10.1109/TEVC.2002.1011539)
5. Herrera-Lozada J.C., Calvo H., Taud H. A Micro Artificial Immune System // Polibits. 2011. Vol. 43. P. 107-111.
6. Карпенко А.П., Шуров Д.Л. Метод глобальной оптимизации SIA. Гибридизация иммунного метода HIA и метода локального поиска SUBPLEX. Saarbrücken: LAMBERT Academic Publishing, 2012. 57 с.
7. D'haeseleer P., Forrest S., Helman P. An immunological approach to change detection: algorithms, analysis and implications // Proceedings of the 1996 IEEE Symposium on Security and Privacy. 1996. P. 110-119. DOI: [10.1109/SECPRI.1996.502674](https://doi.org/10.1109/SECPRI.1996.502674)
8. Castro L.N., Timmis J. An artificial immune network for multimodal function optimization // Proceedings of the 2002 IEEE Congress on Evolutionary Computation (CEC'02). Vol. 1. 2002. P. 699-674. DOI: [10.1109/CEC.2002.1007011](https://doi.org/10.1109/CEC.2002.1007011)
9. Карпенко А.П., Селиверстов Е.Ю. Глобальная оптимизация методом роя частиц. Обзор // Информационные технологии. 2010. № 2. С. 25-34.
10. Черноусько Ф. Л., Болотник Н. Н. Мобильные роботы, управляемые движением внутренних тел // Труды института математики и механики УрО РАН. 2010. Т. 16, № 5. С. 213-222.
11. Болотник Н. Н., Зейдис И. М., Циммерманн К., Яцун С.Ф. Динамика управляемых движений вибрационных систем // Известия РАН. Теория и системы управления. 2006. № 5. С. 157-167.

12. Yap D.F.W., Koh S.P., Tiongand S.K., Prajindra S.K. Particle Swarm based Artificial Immune System for Multimodal Function Optimization and Engineering Application Problem // Trends in Applied Sciences Research. 2011. Vol. 6, no. 3. P. 282-293.
13. Дьяконов В.П. MATLAB и SIMULINK для радиоинженеров. М.: «ДМК-Пресс», 2011. 976 с.
14. Федоренко Р.П. Приближенное решение задач оптимального управления. М.: Наука, 1978. 488 с.

A global optimization hybrid algorithm based on the algorithm of artificial immune system and swarm of particles

03, March 2014

DOI: **10.7463/0314.0702743****A.P. Karpenko, N.O. Shcherbakova, V.A. Bulanov**

Bauman Moscow State Technical University, 105005, Moscow, Russian Federation

apkarpenko@mail.rushcherbakova@gmail.comvb001@rambler.ru

Experience to solve complex problems of global optimization using a meta heuristic algorithms shows that application of one such algorithm not always leads to success. Therefore in recent years much attention is paid to hybridization of meta heuristic algorithms of global optimization. In the hybrid algorithms uniting various algorithms or identical algorithms, but with various values of free parameters, the efficiency of one algorithm can compensate weakness of another.

The paper concerns the hybridization of population algorithm based on the artificial immune system (AIS) and algorithm of particles of swarm (PSO). We use hybridization according to the scheme the pre-processor /post-processor. The AIS and PSO optimization algorithms possess complementary properties. So, it is known that AIS algorithms have high probability of global extremum localization of objective function, but low convergence speed. On the contrary, for PSO algorithms the high speed of convergence is intrinsic, but, if no additional efforts are made, there is rather low probability of global extremum localization. Hybridization of AIS and PSO algorithms pursues the aim to unite the specified advantages of these algorithms and to level their shortcomings.

The paper offers the AIS_PSO algorithm that represents a modification of hybrid algorithm based on the immune algorithm of CLONALG and algorithm of particles of swarm, PSO. Modification consists in replacement of algorithm of CLONALG with the same, but the modified algorithm. The MatLab software complex is used to provide a program implementation of AIS_PSO algorithm. An efficiency of developed algorithm and software has been examined using Rosenbrock's, Himmelblau's, and Rastrigin's test functions. The AIS_PSO software allows a problem solution to provide the optimal control of vibration robot movement.

A known method is used to solve the problem of the optimal control of dynamic system based on reduction of this task to a task of nonlinear programming of high dimension. The study results have shown that a developed algorithm and software are high efficient.

Publications with keywords: [global optimization](#), [PSO](#), [hybrid algorithm](#)

Publications with words: [global optimization](#), [PSO](#), [hybrid algorithm](#)

References

1. Wang X. *Hybrid nature-inspired computation method for optimization. Doctoral Dissertation.* Helsinki University of Technology, TKK Dissertations, 2009, 161 p.
2. El-Abd, Kamel M. A taxonomy of cooperative search algorithms. In: *Hybrid Metaheuristics. Second International Workshop. Vol. 3636.* 2005, pp. 32-41.
3. Raidl G.R. A Unified View on Hybrid Metaheuristics. In: *Hybrid Metaheuristics.* Springer Berlin Heidelberg, 2006, pp. 1-12. (Ser. *Lecture Notes in Computer Science*; vol. 4030). DOI: 10.1007/11890584_1
4. Castro L.N., Zuben F.J. Learning and optimization using the clonal selection principle. *IEEE Transactions on Evolutionary Computation*, 2002, vol. 6, no. 3, pp. 239-251. DOI: [10.1109/TEVC.2002.1011539](#)
5. Herrera-Lozada J.C., Calvo H., Taud H. A Micro Artificial Immune System. *Polibits*, 2011, vol. 43, pp. 107-111.
- Karpenko A.P., Shurov D.L. *Metod global'noy optimizatsii SIA. Gibridizatsiya immunnogo metoda HIA i metoda lokal'nogo poiska SUBPLEX* [Global optimization method SIA. Hybridization immune method HIA, and local search method SUBPLEX]. Saarbrücken, LAMBERT Academic Publishing, 2012. 57 p. (in Russian).
6. D'haeseleer P., Forrest S., Helman P. An immunological approach to change detection: algorithms, analysis and implications. *Proceedings of the 1996 IEEE Symposium on Security and Privacy*, 1996, pp. 110-119. DOI: [10.1109/SECPRI.1996.502674](#)
7. Castro L.N., Timmis J. An artificial immune network for multimodal function optimization. *Proceedings of the 2002 IEEE Congress on Evolutionary Computation (CEC'02). Vol. 1*, 2002, pp. 699-674. DOI: [10.1109/CEC.2002.1007011](#)
8. Karpenko A.P., Seliverstov E.Yu. [Global Optimization by the Particle Swarm Optimization Method. Review]. *Informatsionnye tekhnologii*, 2010, no. 2, pp. 25-34. (in Russian).
9. Chernous'ko F. L., Bolotnik N. N. [Mobile robots, controlled by the movement of internal bodies]. *Trudy instituta matematiki i mekhaniki UrO RAN*, 2010, vol. 16, no. 5, pp. 213-222. (in Russian).
10. Bolotnik N. N., Zeydis I. M., Tsimmermann K., Yatsun S.F. [Dynamics of controlled motion of vibration-driven systems]. *Izvestiya RAN. Teoriya i sistemy upravleniya*, 2006, no. 5, pp. 157-167.

(English translation: *Journal of Computer and Systems Sciences International*, 2006, vol. 45, iss. 5, pp. 831-840. DOI: 10.1134/S1064230706050145).

11. Yap D.F.W., Koh S.P., Tiongand S.K., Prajindra S.K. Particle Swarm based Artificial Immune System for Multimodal Function Optimization and Engineering Application Problem. *Trends in Applied Sciences Research*, 2011, vol. 6, no. 3, pp. 282-293.
12. D'yakonov V.P. *MATLAB i SIMULINK dlya radioinzhenerov* [MATLAB and SIMULINK for radio engineers]. Moscow, DMK-Press, 2011. 976 p. (in Russian).
13. Fedorenko R.P. *Priblizhennoe reshenie zadach optimal'nogo upravleniya* [Approximate solution of optimal control problems]. Moscow, Nauka, 1978. 488 p. (in Russian).