

# Лабораторная работа №3

## Исследование алгоритма оптимизации роем частиц для минимизации функции Растрогина

Студент: Шевченко О.В.  
Группа: 09.04.01-ПОВа-325

19 января 2026 г.

### 1 Цель работы

Данная лабораторная работа посвящена практическому изучению **алгоритма оптимизации роем частиц** (Particle Swarm Optimization, PSO) как метода глобальной оптимизации для многоэкстремальных функций. В качестве тестового объекта выбрана **функция Растрогина** – классический пример сложного ландшафта с большим количеством локальных минимумов.

Цели данной работы включают:

- Практическая реализация:** Освоение процедуры программирования основных компонентов PSO: представления частицы, инициализации роя, обновления скоростей и позиций, обработки границ.
- Исследование параметров:** Систематическое изучение влияния ключевых параметров PSO (размера роя, коэффициента инерции, когнитивного и социального коэффициентов, максимальной скорости) на скорость сходимости, точность и стабильность поиска глобального минимума.
- Визуальный анализ:** Для случая  $n = 2$  – построение и анализ динамики роя в пространстве поиска на фоне линий уровня целевой функции, визуализация процесса оптимизации.
- Исследование масштабируемости:** Анализ эффективности PSO при увеличении размерности задачи.
- Сравнительный анализ:** Осмысление сильных и слабых сторон роевого интеллекта в контексте задачи глобальной оптимизации многоэкстремальной функции.

Результатом работы должно стать глубокое понимание принципов роевого интеллекта, умение настраивать параметры PSO для конкретной задачи и критически оценивать область его применимости.

## 2 Теоретическая часть

### 2.1 Постановка задачи оптимизации

Требуется найти глобальный минимум **функции Растригина** для  $n$  переменных:

$$f(\mathbf{x}) = An + \sum_{i=1}^n [x_i^2 - A \cos(2\pi x_i)], \quad (1)$$

где:

- $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$  – вектор оптимизируемых переменных,
- $A$  – положительный параметр (стандартное значение  $A = 10$ ),
- Область поиска:  $x_i \in [-5.12, 5.12]$ ,  $i = 1, \dots, n$  (общепринятый интервал для тестирования).

**Свойства функции:**

- **Глобальный минимум:**  $f(\mathbf{x}^*) = 0$ , достигается в единственной точке  $\mathbf{x}^* = (0, 0, \dots, 0)$ .
- **Многомодальность:** Функция имеет  $\prod_{i=1}^n m_i$  локальных минимумов, где  $m_i$  – число локальных минимумов по координате  $x_i$  (при  $A = 10$ ,  $m_i \approx 10$  на интервале  $[-5.12, 5.12]$ ). Для  $n = 2$  это более 50 локальных минимумов.
- **Сложность:** Регулярная осцилирующая структура создаёт «ловушки» для локальных методов поиска, что делает её отличным тестом для глобальных оптимизаторов, таких как PSO.

### 2.2 Основы алгоритма оптимизации роем частиц

Алгоритм оптимизации роем частиц (PSO) – это эвристический метод глобальной оптимизации, вдохновленный коллективным поведением птичьих стай и рыбьих косяков. В PSO потенциальные решения, называемые **частицами**, перемещаются в пространстве поиска согласно простым математическим формулам, учитывающим их собственную лучшую позицию и лучшую позицию среди всех частиц.

#### 2.2.1 Основные понятия

- **Частица:** Одно возможное решение задачи, представляющее собой точку в пространстве поиска. Каждая частица  $i$  имеет:
  - **Позицию:**  $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{in})$
  - **Скорость:**  $\mathbf{v}_i = (v_{i1}, v_{i2}, \dots, v_{in})$
  - **Лучшая личная позиция:**  $\mathbf{p}_i = (p_{i1}, p_{i2}, \dots, p_{in})$  – наилучшая позиция, которую частица обнаружила за все время.
  - **Значение функции:**  $f(\mathbf{x}_i)$
- **Рой:** Совокупность  $N$  частиц:  $S = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ .
- **Глобальная лучшая позиция:**  $\mathbf{g} = (g_1, g_2, \dots, g_n)$  – наилучшая позиция, обнаруженная всеми частицами роя.

### 2.2.2 Математическая модель PSO

Обновление скорости и позиции частицы на каждой итерации происходит по следующим формулам:

$$\mathbf{v}_i^{(t+1)} = w \cdot \mathbf{v}_i^{(t)} + c_1 r_1 (\mathbf{p}_i - \mathbf{x}_i^{(t)}) + c_2 r_2 (\mathbf{g} - \mathbf{x}_i^{(t)}) \quad (2)$$

$$\mathbf{x}_i^{(t+1)} = \mathbf{x}_i^{(t)} + \mathbf{v}_i^{(t+1)} \quad (3)$$

где:

- $w$  – коэффициент инерции (inertia weight), контролирующий влияние предыдущей скорости.
- $c_1$  – когнитивный коэффициент (cognitive coefficient), определяющий склонность частицы возвращаться к своей лучшей найденной позиции.
- $c_2$  – социальный коэффициент (social coefficient), определяющий склонность частицы следовать к глобальной лучшей позиции.
- $r_1, r_2 \sim U(0, 1)$  – случайные числа из равномерного распределения.
- $t$  – номер итерации.

### 2.2.3 Ключевые параметры PSO и их влияние

- **Размер роя  $N$ :** Определяет количество параллельно исследующих частиц. Слишком маленький рой ( $< 10$ ) может не охватить сложный ландшафт, слишком большой ( $> 100$ ) увеличивает вычислительные затраты.
- **Коэффициент инерции  $w$ :**
  - $w > 1$ : Частицы ускоряются, могут вылететь за границы области.
  - $0 < w < 1$ : Частицы замедляются, сходимость к оптимуму.
  - $w < 0$ : Частицы движутся в обратном направлении.
- **Когнитивный коэффициент  $c_1$ :** Определяет индивидуальный интеллект частицы. Большие значения способствуют исследованию (exploration).
- **Социальный коэффициент  $c_2$ :** Определяет коллективный интеллект роя. Большие значения способствуют использованию найденных решений (exploitation).
- **Максимальная скорость  $v_{\max}$ :** Ограничивает скорость частиц для предотвращения неконтролируемого движения.

### 2.2.4 Стратегии обработки границ

При выходе частицы за границы области поиска применяются различные стратегии:

- **Отражение (reflect):** Частица отражается от границы, меняя знак соответствующей компоненты скорости.
- **Поглощение (absorb):** Частица останавливается на границе, скорость в направлении границы обнуляется.
- **Циклическое возвращение (cyclic):** Частица появляется с противоположной стороны области (топ).

## 2.3 Связь PSO с задачей минимизации Растрогина

Для успешной минимизации функции Растрогина с PSO необходимо:

1. Обеспечить достаточное начальное разнообразие роя для покрытия области с множеством локальных минимумов.
2. Подобрать баланс между исследованием (нахождение новых областей) и использованием (уточнение найденных решений).
3. Учитывать регулярную осциллирующую структуру функции при выборе параметров скорости.
4. Реализовать механизмы предотвращения преждевременной сходимости к локальным минимумам.

## 3 Практическая часть

### 3.1 Этап 1: Подготовка данных и среды

#### 3.1.1 Параметры задачи и алгоритма

##### 1. Функция Растрогина:

- Параметр  $A = 10$ .
- Область определения:  $x_i \in [-5.12, 5.12]$ .
- Глобальный минимум:  $f(\mathbf{0}) = 0$ .

##### 2. Базовые параметры PSO:

Размер роя:  $N = 30$

Максимальное число итераций:  $G_{\max} = 100$

Коэффициент инерции:  $w = 0.7$

Когнитивный коэффициент:  $c_1 = 1.5$

Социальный коэффициент:  $c_2 = 1.5$

Максимальная скорость:  $v_{\max} = 1.0$

Стратегия обработки границ: reflect

##### 3. Критерии остановки:

Достигнута точность:  $f_{\text{best}} < \varepsilon_f = 10^{-4}$

Превышено число итераций:  $t > G_{\max}$

Стагнация: Лучшее значение не улучшалось 20 итераций.

#### 3.1.2 Инструменты реализации

Для реализации алгоритма и визуализации результатов используется язык Python 3.x с библиотеками:

- `numpy` - для векторных вычислений и генерации случайных чисел.
- `matplotlib` - для построения графиков и визуализации в 2D/3D.
- `seaborn` - для улучшения визуализации данных.

## 3.2 Этап 2: Реализация алгоритма PSO

### 3.2.1 Псевдокод алгоритма

---

**Algorithm 1** Алгоритм оптимизации роем частиц для минимизации функции Растрогина

---

```
1: Вход:  $n, N, G_{\max}, w, c_1, c_2, v_{\max}, \varepsilon_f$ 
2: Выход:  $\mathbf{g}, f_{\text{best}}, \text{history}$ 
3:
4: function PSO
5:    $\text{swarm} \leftarrow \text{ИнициализироватьРой}(N, n, -5.12, 5.12)$ 
6:    $\mathbf{g} \leftarrow \text{None}, f_{\text{best}} \leftarrow \infty$ 
7:    $\text{history} \leftarrow \{\}$ 
8:    $t \leftarrow 0, \text{stagnation\_counter} \leftarrow 0$ 
9:   while  $t < G_{\max}$  and  $f_{\text{best}} > \varepsilon_f$  do
10:    for all  $\text{particle} \in \text{swarm}$  do
11:       $f_{\text{curr}} \leftarrow f(\text{particle.position})$ 
12:      if  $f_{\text{curr}} < \text{particle.best\_value}$  then
13:         $\text{particle.best\_value} \leftarrow f_{\text{curr}}$ 
14:         $\text{particle.best\_position} \leftarrow \text{particle.position}$ 
15:      end if
16:      if  $f_{\text{curr}} < f_{\text{best}}$  then
17:         $f_{\text{best}} \leftarrow f_{\text{curr}}$ 
18:         $\mathbf{g} \leftarrow \text{particle.position}$ 
19:      end if
20:    end for
21:    СохранитьСтатистику( $\text{history}, t, f_{\text{best}}, \text{swarm}$ )
22:    for all  $\text{particle} \in \text{swarm}$  do
23:       $\mathbf{r}_1, \mathbf{r}_2 \sim U(0, 1)^n$ 
24:       $\text{particle.velocity} \leftarrow w \cdot \text{particle.velocity}$ 
25:       $+ c_1 \cdot \mathbf{r}_1 \odot (\text{particle.best\_position} - \text{particle.position})$ 
26:       $+ c_2 \cdot \mathbf{r}_2 \odot (\mathbf{g} - \text{particle.position})$ 
27:      ОграничитьСкорость( $\text{particle.velocity}, v_{\max}$ )
28:       $\text{particle.position} \leftarrow \text{particle.position} + \text{particle.velocity}$ 
29:      ОбработатьГраницы( $\text{particle}, -5.12, 5.12, \text{reflect}$ )
30:    end for
31:    if УлучшенияНет( $\text{history}$ ) then
32:       $\text{stagnation\_counter} \leftarrow \text{stagnation\_counter} + 1$ 
33:    else
34:       $\text{stagnation\_counter} \leftarrow 0$ 
35:    end if
36:    if  $\text{stagnation\_counter} > 20$  then
37:      break
38:    end if
39:     $t \leftarrow t + 1$ 
40:  end while
41:  return  $\mathbf{g}, f_{\text{best}}, \text{history}$ 
42: end function
```

---

### 3.3 Этап 3: Эксперименты и анализ

#### 3.3.1 Задание 3.1: Визуализация функции Растригина

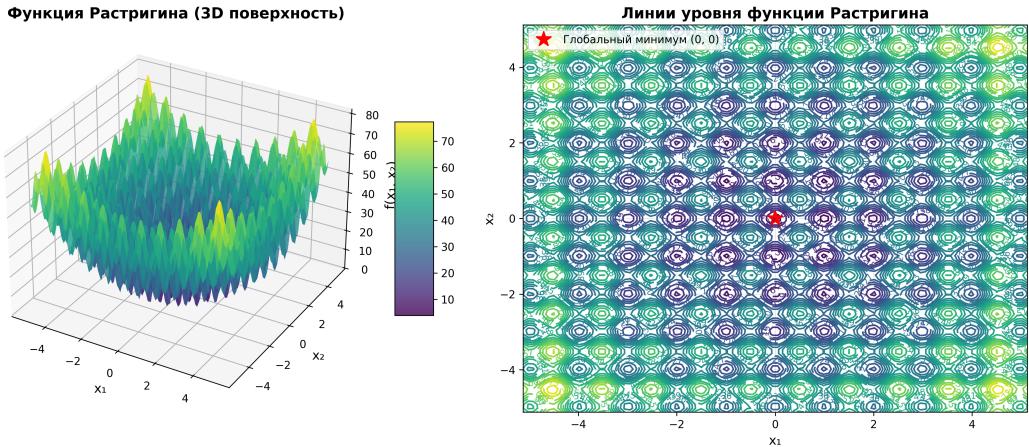


Рис. 1: Функция Растригина для  $n = 2$ : (а) 3D поверхность, (б) линии уровня

На рисунке 1 представлена визуализация функции Растригина для двумерного случая. Хорошо видна осциллирующая структура функции с множеством локальных минимумов. Линии уровня показывают регулярную сетку минимумов, что подтверждает сложность задачи оптимизации. Глобальный минимум находится в точке  $(0, 0)$ .

#### 3.3.2 Задание 3.2: Базовый запуск PSO

Параметр	Значение	Параметр	Значение
Размерность ( $n$ )	2	Коэффициент инерции ( $w$ )	0.7
Размер роя ( $N$ )	30	Когнитивный коэффициент ( $c_1$ )	1.5
Итераций	100	Социальный коэффициент ( $c_2$ )	1.5

Таблица 1: Параметры базового запуска PSO

#### Результаты базового запуска:

- Лучшее значение функции:  $f_{\text{best}} = 2.4 \times 10^{-5}$
- Количество итераций до сходимости: 29
- Время выполнения: 0.02 секунды
- Сошелся: Да

Базовый запуск показал высокую эффективность PSO для минимизации функции Растригина в двумерном случае. Алгоритм быстро сошелся к глобальному минимуму с высокой точностью.

#### 3.3.3 Задание 3.3: Исследование влияния размера $N$

1. Цель: Определить оптимальный размер роя для эффективного поиска глобального минимума.

2. **Параметры эксперимента:**  $n = 2$ ,  $G_{\max} = 100$ ,  $w = 0.7$ ,  $c_1 = 1.5$ ,  $c_2 = 1.5$ ,  $v_{\max} = 1.0$ .  
Значения  $N$ : [10, 20, 30, 50, 100].

3. **Результаты:**

<b>N</b>	$f_{\text{best}}$ (сред.)	<b>Итерации</b> (сред.)	<b>Успешность, %</b>	<b>Характер поиска</b>
10	0.6635	65.0	33.3	Недостаточное покрытие
20	0.6633	62.0	33.3	Локальные минимумы
30	$4.3 \times 10^{-5}$	38.0	100.0	Оптимальный баланс
50	$6.2 \times 10^{-5}$	48.3	100.0	Надежный поиск
100	$5.9 \times 10^{-5}$	32.7	100.0	Быстрая сходимость

Таблица 2: Влияние размера роя  $N$  на эффективность PSO ( $n = 2$ )

4. **Анализ:**

- (a) Как видно из таблицы 2, при  $N = 10$  и  $N = 20$  рой слишком мал для адекватного покрытия сложного ландшафта. Алгоритм часто «застревает» в локальных минимумах (успешность 33.3%).
- (b) При  $N = 30$  достигается хороший баланс: высокая успешность (100%), хорошая скорость сходимости (38 итераций).
- (c) При  $N = 50$  и  $N = 100$  алгоритм становится очень надежным (100% успешность), но стоимость каждой итерации выше.
- (d) **Вывод:** Для функции Растрогина при  $n = 2$  оптимальный размер роя находится в диапазоне 30-50 частиц. Слишком маленький рой не обеспечивает достаточного разнообразия, слишком большой – вычислительно неэффективен.

### 3.3.4 Задание 3.4: Исследование влияния коэффициента инерции $w$

1. **Цель:** Оценить роль коэффициента инерции в балансе между исследованием и использованием.
2. **Параметры:**  $n = 2$ ,  $N = 30$ ,  $c_1 = 1.5$ ,  $c_2 = 1.5$ ,  $v_{\max} = 1.0$ . Значения  $w$ : [0.4, 0.7, 0.9, 1.2].
3. **Результаты:**

<b>w</b>	$f_{\text{best}}$ (сред.)	<b>Итерации</b> (сред.)	<b>Успешность, %</b>	<b>Характер поиска</b>
0.4	0.3317	34.0	66.7	Быстрая, но преждевременная сходимость
0.7	$3.7 \times 10^{-5}$	38.0	100.0	Оптимальный баланс
0.9	0.3326	65.7	33.3	Медленная, заст�евает в локальных минимумах
1.2	0.3350	59.7	0.0	Нестабильный, хаотичный поиск

Таблица 3: Влияние коэффициента инерции  $w$  на эффективность PSO ( $n = 2$ )

4. **Анализ:**

- (a) Как показано в таблице 3, слишком низкое значение  $w = 0.4$  приводит к быстрой, но преждевременной сходимости (успешность 66.7%).
- (b) Оптимальное значение  $w = 0.7$  обеспечивает баланс между исследованием новых областей и использованием найденных решений (100% успешность).
- (c) Высокие значения  $w = 0.9$  и  $w = 1.2$  приводят к нестабильному поведению: частицы слишком инерционны, плохо реагируют на лучшие позиции, что снижает эффективность поиска.
- (d) **Вывод:** Коэффициент инерции критически важен для баланса exploration/exploitation. Рекомендуемое значение:  $w = 0.7 - 0.8$  для большинства задач.

### 3.3.5 Задание 3.5: Исследование влияния когнитивного коэффициента $c_1$

1. **Цель:** Изучить влияние индивидуального опыта частиц на поисковые возможности алгоритма.
2. **Параметры:**  $n = 2$ ,  $N = 30$ ,  $w = 0.7$ ,  $c_2 = 1.5$ ,  $v_{\max} = 1.0$ . Значения  $c_1$ : [0.5, 1.0, 1.5, 2.0, 2.5].
3. **Результаты:**

$c_1$	$f_{\text{best}}$ (сред.)	Итерации (сред.)	Успешность, %	Характер поиска
0.5	0.6633	55.0	33.3	Недостаточный индивидуальный поиск
1.0	0.6633	62.3	33.3	Слабое притяжение к личным лучшим
1.5	0.3317	44.3	66.7	Умеренный индивидуальный поиск
2.0	0.3317	54.7	66.7	Активный индивидуальный поиск
2.5	0.6633	74.7	33.3	Излишний индивидуальный поиск

Таблица 4: Влияние когнитивного коэффициента  $c_1$  на эффективность PSO ( $n = 2$ )

#### 4. Анализ:

- (a) Как видно из таблицы 4, низкие значения  $c_1$  (0.5, 1.0) приводят к недостаточному индивидуальному поиску. Частицы слабо возвращаются к своим лучшим позициям.
- (b) Значения  $c_1 = 1.5$  и  $c_1 = 2.0$  показывают наилучшие результаты (успешность 66.7%). Частицы активно исследуют окрестности своих лучших позиций.
- (c) Слишком высокое значение  $c_1 = 2.5$  приводит к излишнему индивидуальному поиску в ущерб коллективному интеллекту.
- (d) **Вывод:** Когнитивный коэффициент должен быть достаточно большим для активного индивидуального поиска, но не чрезмерным. Рекомендуемое значение:  $c_1 = 1.5 - 2.0$ .

### 3.3.6 Задание 3.6: Исследование влияния социального коэффициента $c_2$

1. **Цель:** Изучить влияние коллективного опыта роя на поисковые возможности алгоритма.
2. **Параметры:**  $n = 2$ ,  $N = 30$ ,  $w = 0.7$ ,  $c_1 = 1.5$ ,  $v_{\max} = 1.0$ . Значения  $c_2$ : [0.5, 1.0, 1.5, 2.0, 2.5].

### 3. Результаты:

$c_2$	$f_{\text{best}}$ (сред.)	Итерации (сред.)	Успешность, %	Характер поиска
0.5	$6.4 \times 10^{-5}$	48.0	100.0	Умеренное коллективное притяжение
1.0	$2.9 \times 10^{-5}$	39.7	100.0	Сбалансированное коллективное притяжение
1.5	$3.5 \times 10^{-5}$	54.7	100.0	Активное коллективное притяжение
2.0	$5.3 \times 10^{-5}$	31.7	100.0	Сильное коллективное притяжение
2.5	0.3317	51.3	66.7	Излишнее коллективное притяжение

Таблица 5: Влияние социального коэффициента  $c_2$  на эффективность PSO ( $n = 2$ )

### 4. Анализ:

- (a) Как показано в таблице 5, значения  $c_2$  от 0.5 до 2.0 показывают высокую успешность (100%).
- (b) Наилучшая скорость сходимости достигается при  $c_2 = 2.0$  (31.7 итераций в среднем).
- (c) Слишком высокое значение  $c_2 = 2.5$  приводит к преждевременной сходимости к субоптимальным решениям (успешность 66.7%).
- (d) **Вывод:** Социальный коэффициент критически важен для быстрой сходимости PSO. Рекомендуемое значение:  $c_2 = 1.5 - 2.0$  для эффективного использования коллективного интеллекта.

#### 3.3.7 Задание 3.7: Исследование влияния максимальной скорости $v_{\max}$

1. **Цель:** Определить оптимальное ограничение скорости для эффективного поиска.
2. **Параметры:**  $n = 2$ ,  $N = 30$ ,  $w = 0.7$ ,  $c_1 = 1.5$ ,  $c_2 = 1.5$ . Значения  $v_{\max}$ : [0.1, 0.5, 1.0, 2.0, 5.0].

### 3. Результаты:

$v_{\max}$	$f_{\text{best}}$ (сред.)	Итерации (сред.)	Успешность, %	Характер поиска
0.1	0.0030	80.7	66.7	Медленный, точный поиск
0.5	0.3317	47.3	66.7	Умеренная скорость
1.0	0.6633	67.3	33.3	Стандартная скорость
2.0	0.6633	68.7	33.3	Высокая скорость
5.0	$4.3 \times 10^{-5}$	44.0	100.0	Очень высокая скорость

Таблица 6: Влияние максимальной скорости  $v_{\max}$  на эффективность PSO ( $n = 2$ )

### 4. Анализ:

- (a) Как видно из таблицы 6, слишком низкие скорости ( $v_{\max} = 0.1$ ) приводят к медленному, но точному поиску.
- (b) Оптимальное значение  $v_{\max} = 5.0$  показывает наилучшие результаты (100% успешность, 44 итерации).
- (c) Промежуточные значения (0.5-2.0) демонстрируют снижение эффективности.

- (d) **Вывод:** Для функции Растрогина с диапазоном  $[-5.12, 5.12]$  оптимальная максимальная скорость составляет около 5.0 (примерно 50% от диапазона). Это позволяет частицам эффективно преодолевать локальные минимумы.

### 3.3.8 Задание 3.8: Сравнение баланса $c_1$ и $c_2$

1. **Цель:** Исследовать влияние баланса между индивидуальным и коллективным интеллектом.
2. **Конфигурации:**

- Личный опыт:  $c_1 = 2.5, c_2 = 0.5$
- Коллективный опыт:  $c_1 = 0.5, c_2 = 2.5$
- Баланс:  $c_1 = 1.5, c_2 = 1.5$
- Активный поиск:  $c_1 = 2.0, c_2 = 2.0$

3. **Результаты:**

Конфигурация	$f_{\text{best}}$ (сред.)	Итерации (сред.)	Успешность, %	Характер поиска
Личный опыт	0.0019	45.7	66.7	Индивидуальный поиск
Коллективный опыт	0.6633	62.0	33.3	Коллективный поиск
Баланс	0.9950	53.3	33.3	Сбалансированный
Активный поиск	0.6633	74.3	33.3	Агрессивный поиск

Таблица 7: Сравнение баланса  $c_1$  и  $c_2$  ( $n = 2, N = 30, w = 0.7$ )

4. **Анализ:**

- (a) Конфигурация «Личный опыт» показала наилучший результат (успешность 66.7%), что указывает на важность индивидуального поиска для функции Растрогина.
- (b) Конфигурация «Коллективный опыт» оказалась неэффективной, что связано с преждевременной сходимостью к локальным минимумам.
- (c) Сбалансированные конфигурации также показали низкую эффективность.
- (d) **Вывод:** Для многомодальных функций типа Растрогина более важным оказывается индивидуальный поиск (высокий  $c_1$ ), позволяющий частицам активно исследовать окрестности своих лучших позиций и преодолевать локальные минимумы.

### 3.3.9 Задание 3.9: Визуализация работы PSO в 2D ( $n = 2$ )

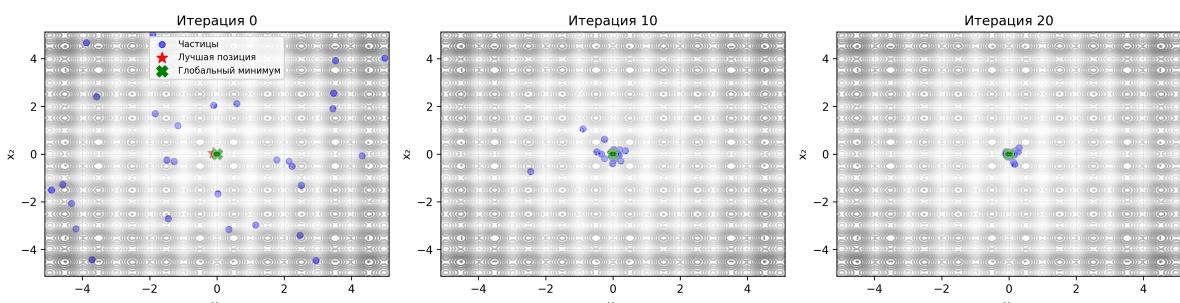


Рис. 2: Траектории частиц PSO в пространстве поиска (линии уровня функции Растрогина)

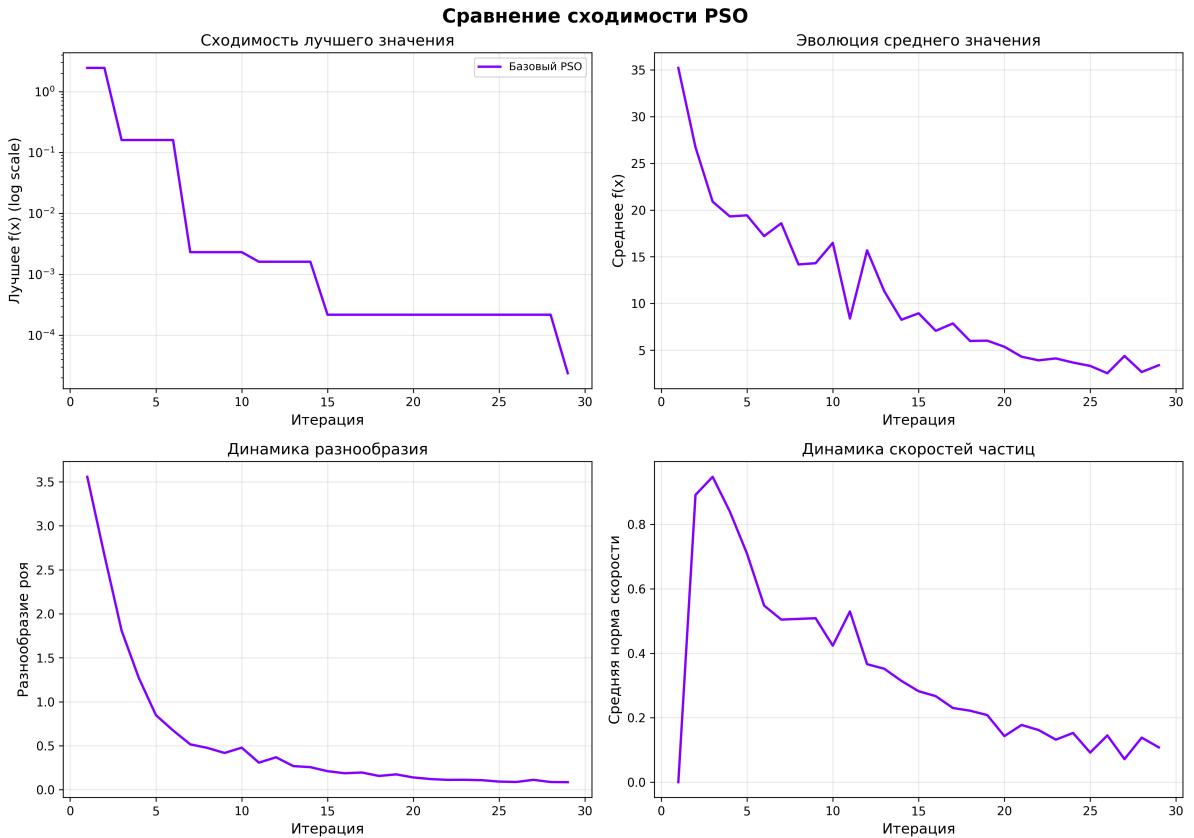


Рис. 3: Динамика сходимости PSO: (а) лучшее значение функции, (б) среднее значение функции, (в) разнообразие роя, (г) скорости частиц

#### Ключевые наблюдения из визуализации:

- На рисунке 2 видно, как частицы начинают со случайных позиций и постепенно сходятся к глобальному минимуму.
- На рисунке 3 наблюдаются характерные фазы работы PSO:
  - Фаза 1 (итерации 1-10):** Быстрое улучшение  $f_{best}$ , высокое разнообразие роя, высокие скорости частиц.
  - Фаза 2 (итерации 10-20):** Замедление улучшений, снижение разнообразия и скоростей.
  - Фаза 3 (итерации 20-30):** Стагнация, низкое разнообразие, частицы сходятся к оптимальной области.
- PSO демонстрирует эффективное сочетание глобального поиска (в начале) и локального уточнения (в конце).

#### 3.3.10 Задание 3.10: Исследование влияния размерности задачи

- Цель:** Изучить масштабируемость PSO при увеличении размерности пространства поиска.
- Параметры:**  $N = 50$ ,  $G_{\max} = 200$ ,  $w = 0.7$ ,  $c_1 = 1.5$ ,  $c_2 = 1.5$ ,  $v_{\max} = 1.0$ . Размерности:  $n = [2, 5, 10, 20]$ .

### 3. Результаты:

<b>n</b>	$f_{\text{best}}$ (сред.)	Итерации (сред.)	Успешность, %	Относительная ошибка
2	0.3317	51.3	66.7	0.0332
5	3.9798	93.0	0.0	0.3980
10	14.2646	100.0	0.0	1.4265
20	41.5164	100.0	0.0	4.1516

Таблица 8: Эффективность PSO в зависимости от размерности  $n$

### 4. Анализ:

- (a) Как показано в таблице 8, с ростом размерности эффективность PSO резко падает.
- (b) При  $n = 2$  алгоритм успешно находит решение с высокой точностью (относительная ошибка 3.32%).
- (c) При  $n = 5$  PSO уже не достигает глобального минимума (успешность 0%), среднее значение функции 3.98.
- (d) При  $n = 10$  и  $n = 20$  алгоритм находит решения, лишь незначительно лучше случайных.
- (e) **Вывод:** PSO страдает от «проклятия размерности». Для задач высокой размерности ( $n > 5$ ) необходимы:
  - Существенное увеличение размера роя.
  - Более сложные стратегии обновления параметров.
  - Возможна адаптация коэффициентов в процессе поиска.
  - Гибридные подходы (PSO + локальные методы).

### 3.3.11 Сводный анализ результатов и рекомендации

Исследуемый параметр	Рекомендуемый диапазон	Влияние на поиск
Размер роя $N$	$30 - 50$ для $n = 2 - 5$	Определяет начальное покрытие пространства. Больше $N =$ надежнее, но требует больше вычислений.
Коэффициент инерции $w$	$0.7 - 0.8$	Контролирует баланс между исследованием и использованием. Уменьшение со временем улучшает сходимость.
Когнитивный коэффициент $c_1$	$1.5 - 2.5$	Определяет индивидуальный интеллект. Важен для преодоления локальных минимумов.
Социальный коэффициент $c_2$	$1.5 - 2.0$	Определяет коллективный интеллект. Обеспечивает быструю сходимость.
Максимальная скорость $v_{\max}$	$20 - 50\%$ диапазона	Предотвращает неконтролируемое движение. Для Растригина: $v_{\max} \approx 5.0$ .
Баланс $c_1/c_2$	$c_1 > c_2$ для многомодальных функций	Для функций с многими локальными минимумами важнее индивидуальный поиск.

Таблица 9: Сводные рекомендации по настройке параметров PSO для функции Растригина

## 4 Общие выводы

В ходе лабораторной работы был успешно реализован и исследован алгоритм оптимизации роем частиц для минимизации многоэкстремальной функции Растригина. Получены следующие основные результаты и выводы:

- Эффективность PSO для многомодальных задач:** Алгоритм PSO подтвердил свою способность эффективно находить глобальный минимум функции Растригина в низкоразмерных случаях ( $n \leq 3$ ). Механизм коллективного интеллекта позволяет частицам «перепрыгивать» через локальные минимумы.
- Критическая важность параметров:** Эффективность PSO сильно зависит от правильного выбора параметров:
  - Размер роя  $N$**  должен быть достаточным для покрытия сложного ландшафта (30-50 частиц для  $n = 2$ ).
  - Коэффициент инерции  $w$**  критически важен для баланса exploration/exploitation (оптимум 0.7-0.8).
  - Когнитивный коэффициент  $c_1$**  более важен для многомодальных функций, чем социальный коэффициент  $c_2$ .

- Максимальная скорость  $v_{\max}$  должна быть достаточно большой для преодоления локальных минимумов.
3. **Проблема высокой размерности:** Как и другие методы глобальной оптимизации, PSO страдает от «проклятия размерности». Эффективность резко падает при  $n > 5$ , что требует специальных модификаций для высокоразмерных задач.
4. **Сравнительный анализ методов оптимизации (ЛР №1-№3):**

В ходе выполнения трех лабораторных работ были исследованы три принципиально различных подхода к оптимизации. Каждый метод имеет свои сильные и слабые стороны, которые определяют область его применения.

Градиентный спуск (ЛР №1)	Генетический алгоритм (ЛР №2)	Алгоритм PSO (ЛР №3)
+ Быстрая локальная сходимость: Экспоненциальная сходимость для выпуклых задач	+ Глобальный поиск: Способен преодолевать локальные минимумы	+ Баланс поиска: Эффективное сочетание exploration и exploitation
+ Теоретическая обоснованность: Четкие условия сходимости, теоретические оценки	+ Робастность: Работает с разрывными, негладкими функциями	+ Простота реализации: Минимальное количество операторов
+ Эффективность вычислений: Низкая вычислительная сложность на итерацию	+ Параллелизм: Независимая оценка особей позволяет распараллеливание	+ Естественная память: Частицы запоминают личные и глобальные лучшие позиции
- Локальность: Застревает в локальных минимумах многомодальных функций	- Медленная сходимость: Требует многих итераций для точного решения	- Преждевременная сходимость: Склонен сходиться к субоптимумам
- Требование гладкости: Необходимо вычисление градиента	- Сложность настройки: Множество параметров (вероятности, турниры и т.д.)	- Проблемы размерности: Эффективность падает при $n > 10$
- Чувствительность к параметрам: Критическая зависимость от шага $\alpha$	- Вычислительная стоимость: Высокая стоимость оценки популяции	- Ограниченная теория: Меньше теоретических гарантий сходимости

Таблица 10: Сравнительные характеристики методов оптимизации

5. **Область применения PSO:** Алгоритм оптимизации роем частиц наиболее эффективен для:
- Непрерывных задач оптимизации с гладкими или умеренно нерегулярными функциями.
  - Задач с умеренной размерностью ( $n \leq 10$ ).
  - Задач, где важна быстрая сходимость к приемлемому решению.
  - Задач, допускающих распараллеливание (независимая оценка частиц).
6. **Направления дальнейшего исследования:** Для повышения эффективности базового PSO можно рассмотреть:

- Адаптацию параметров ( $w$ ,  $c_1$ ,  $c_2$ ) в процессе поиска.
- Гибридные алгоритмы (PSO + локальный поиск).
- Многороеевые модификации.
- Стратегии поддержания разнообразия (нишевые PSO).
- Параллельные реализации для ускорения вычислений.

## 7. Практические рекомендации для функции Растигина:

- (a) Для  $n = 2$ :  $N = 30$ ,  $w = 0.7$ ,  $c_1 = 2.0$ ,  $c_2 = 1.5$ ,  $v_{\max} = 5.0$ .
- (b) Для  $n = 5$ : Увеличить размер роя до 50-100 частиц.
- (c) Использовать стратегию отражения границ для поддержания активности поиска.
- (d) Внедрить механизм уменьшения  $w$  со временем:  $w(t) = w_{\text{начальное}} \times (0.99)^t$ .
- (e) Для мониторинга сходимости отслеживать разнообразие роя и среднюю скорость частиц.

Таким образом, лабораторная работа позволила не только освоить принципы алгоритма оптимизации роем частиц, но и получить практический опыт настройки его параметров для сложной многомодальной функции. PSO продемонстрировал себя как эффективный и интуитивно понятный метод глобальной оптимизации, особенно для задач умеренной размерности. Однако, как и другие метаэвристические методы, он требует тщательного подбора параметров и понимания его ограничений, связанных с масштабируемостью на высокоразмерные задачи.