

# Оптимизация с помощью алгоритма роя частиц

Евгений Ильин

# Обо мне и контакты

## Евгений Ильин

МАИ, Институт №4 «Радиоэлектроника,  
инфокоммуникации и информационная безопасность»

E-mail: [jenyay.ilin@gmail.com](mailto:jenyay.ilin@gmail.com)

Сайт: <https://jenyay.net>

Github: <https://github.com/jenyay>

Телеграм: [@jenyay](https://t.me/jenyay)

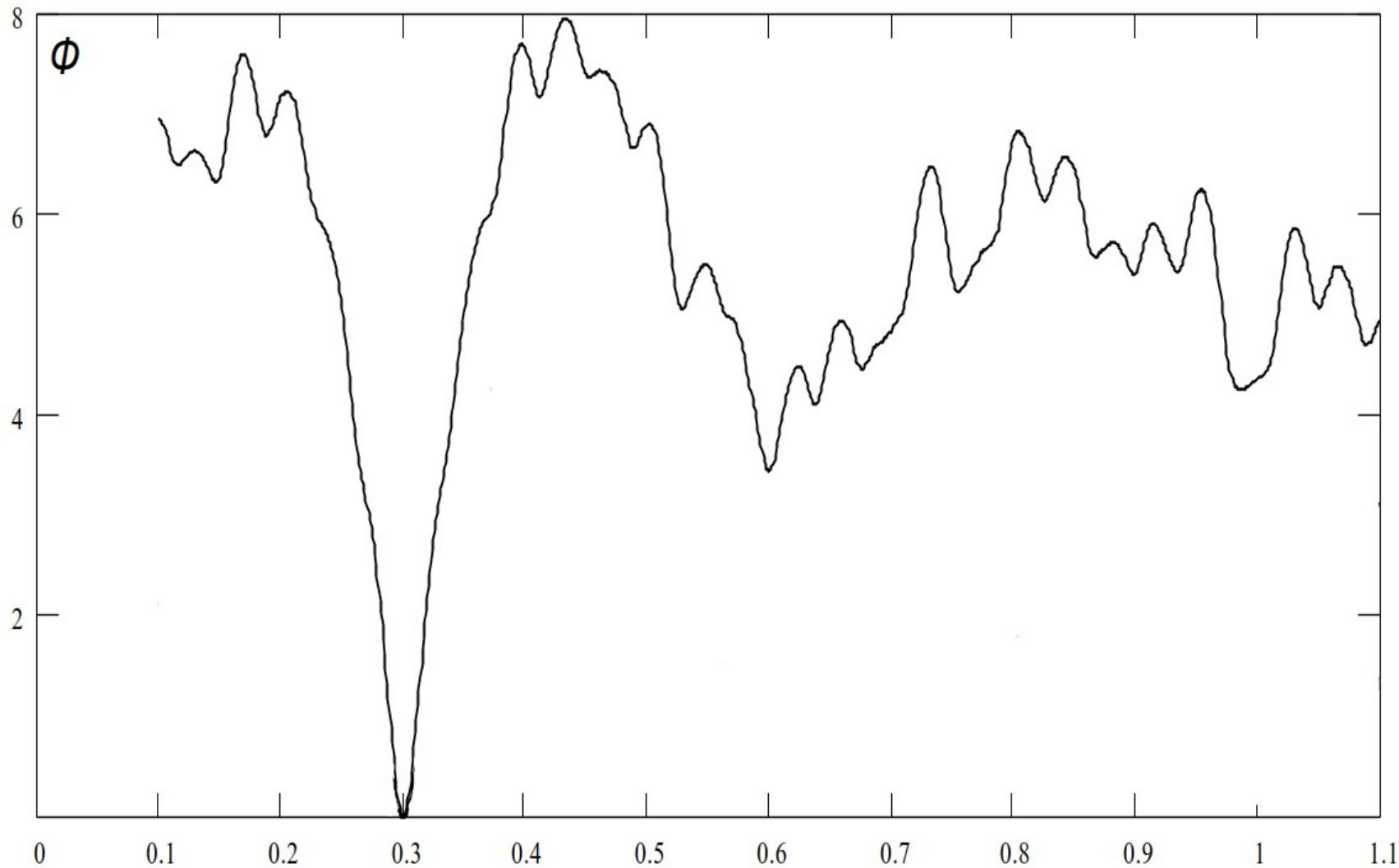
VK: <https://vk.com/jenyay>

FB: <https://www.facebook.com/jenyay.ilin>

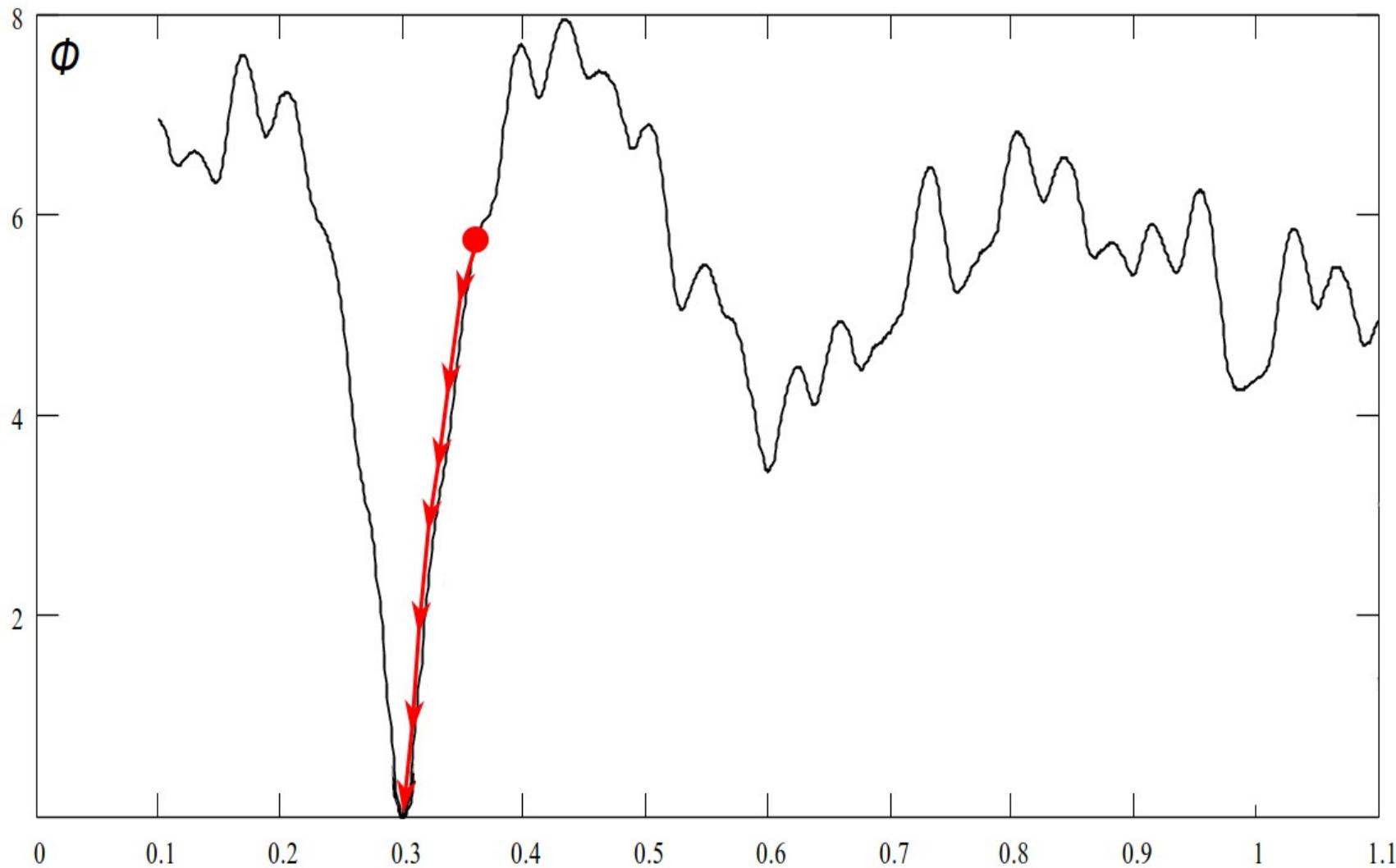
# Оптимизация

**Оптимизация** — (от лат. *optimus* — наилучший) задача нахождения экстремума (максимума или минимума) целевой функции в некоторой области конечномерного векторного пространства, ограниченной набором линейных и/или нелинейных равенств и/или неравенств.

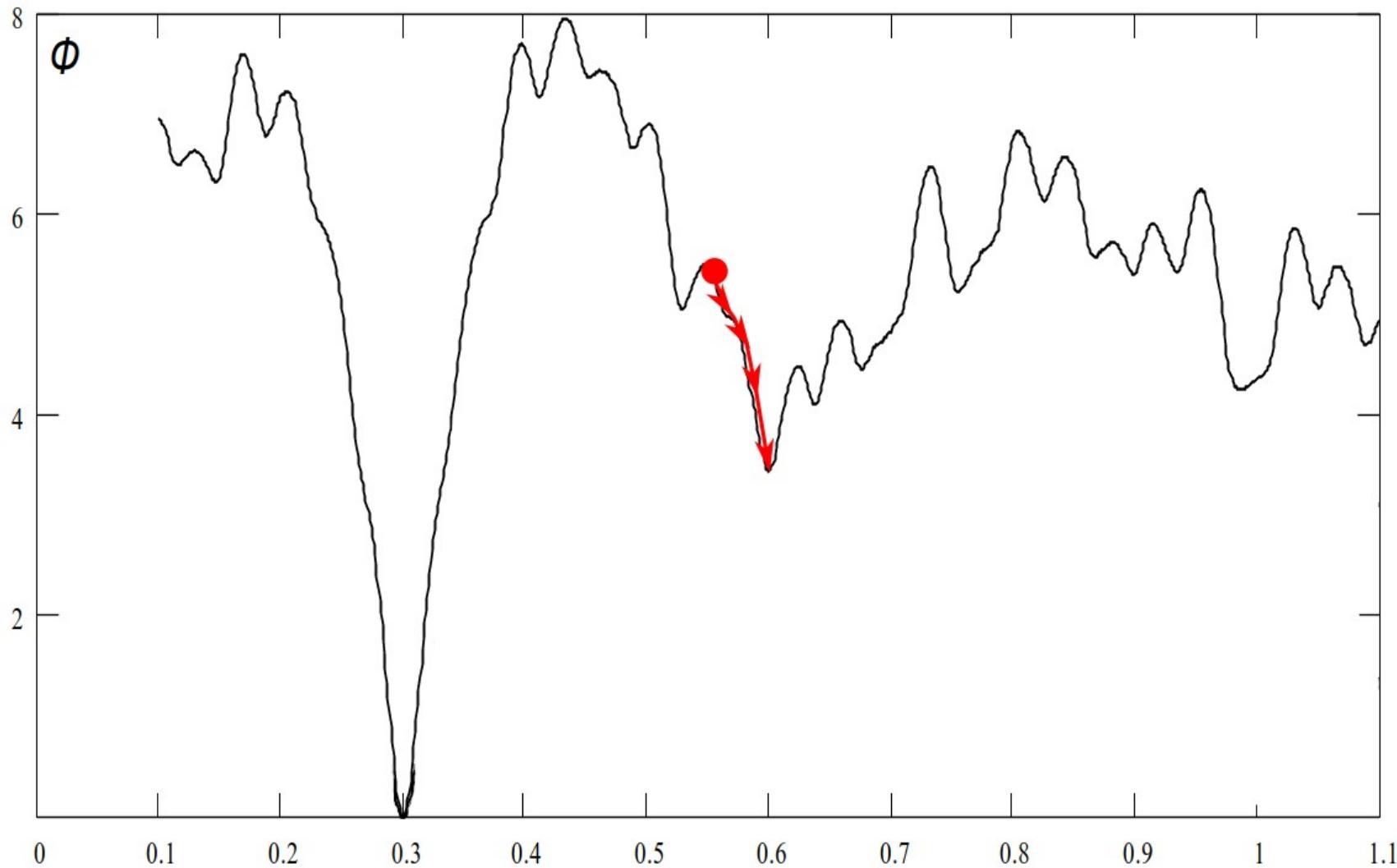
# Примеры целевых функций



# Сходимость градиентного метода



# Сходимость градиентного метода



# Алгоритмы глобальной оптимизации

- Генетический алгоритм
- **Алгоритм роя частиц**
- Метод Нелдера-Мида
- Алгоритм случайного поиска
- Алгоритм имитации отжига
- Алгоритм колонии искусственных пчел
- И многие другие

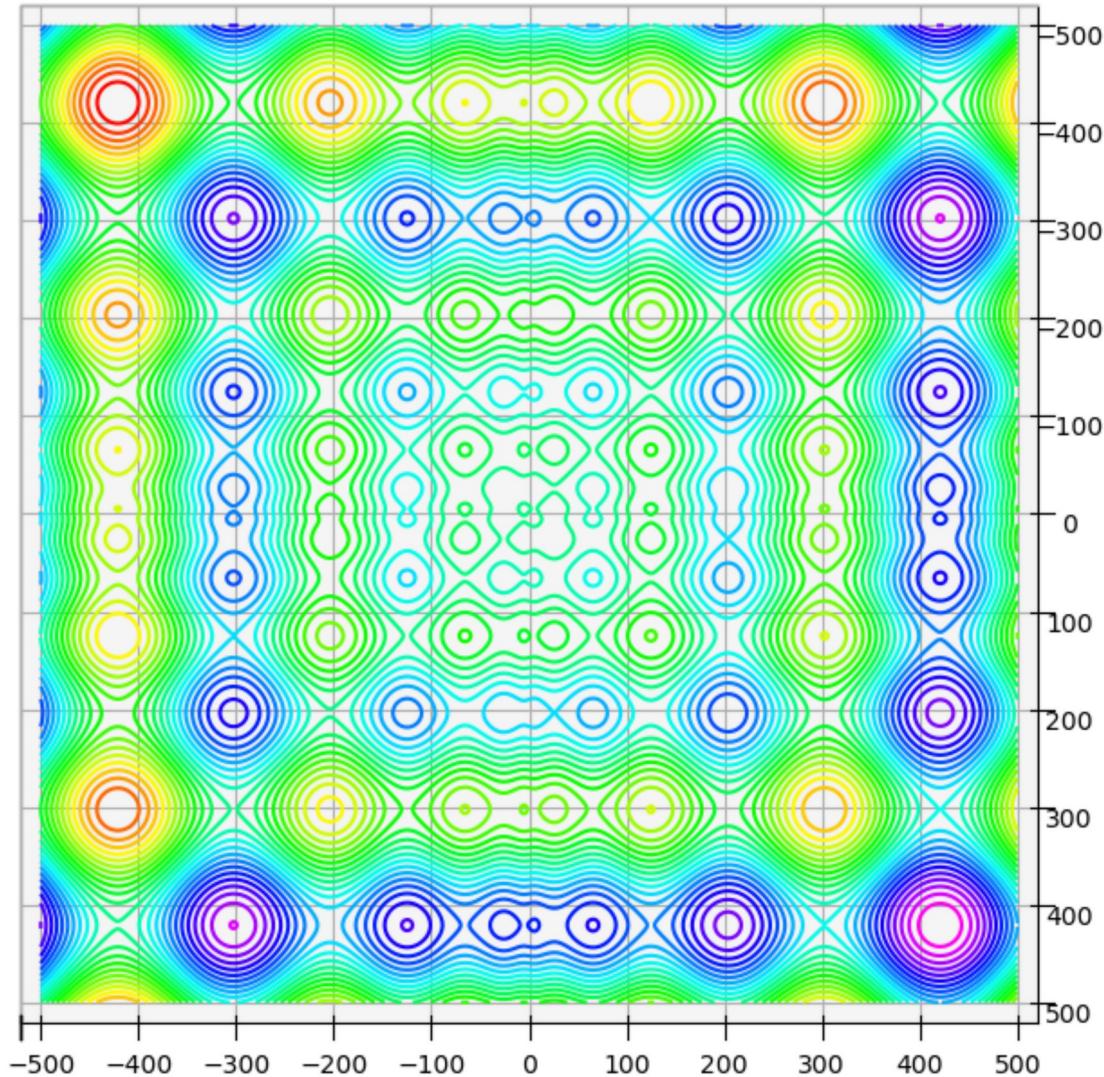
# Алгоритм роя частиц

Kennedy J., Eberhart R. "Particle Swarm Optimization".  
Proceedings of IEEE International Conference  
on Neural Networks. IV. 1995

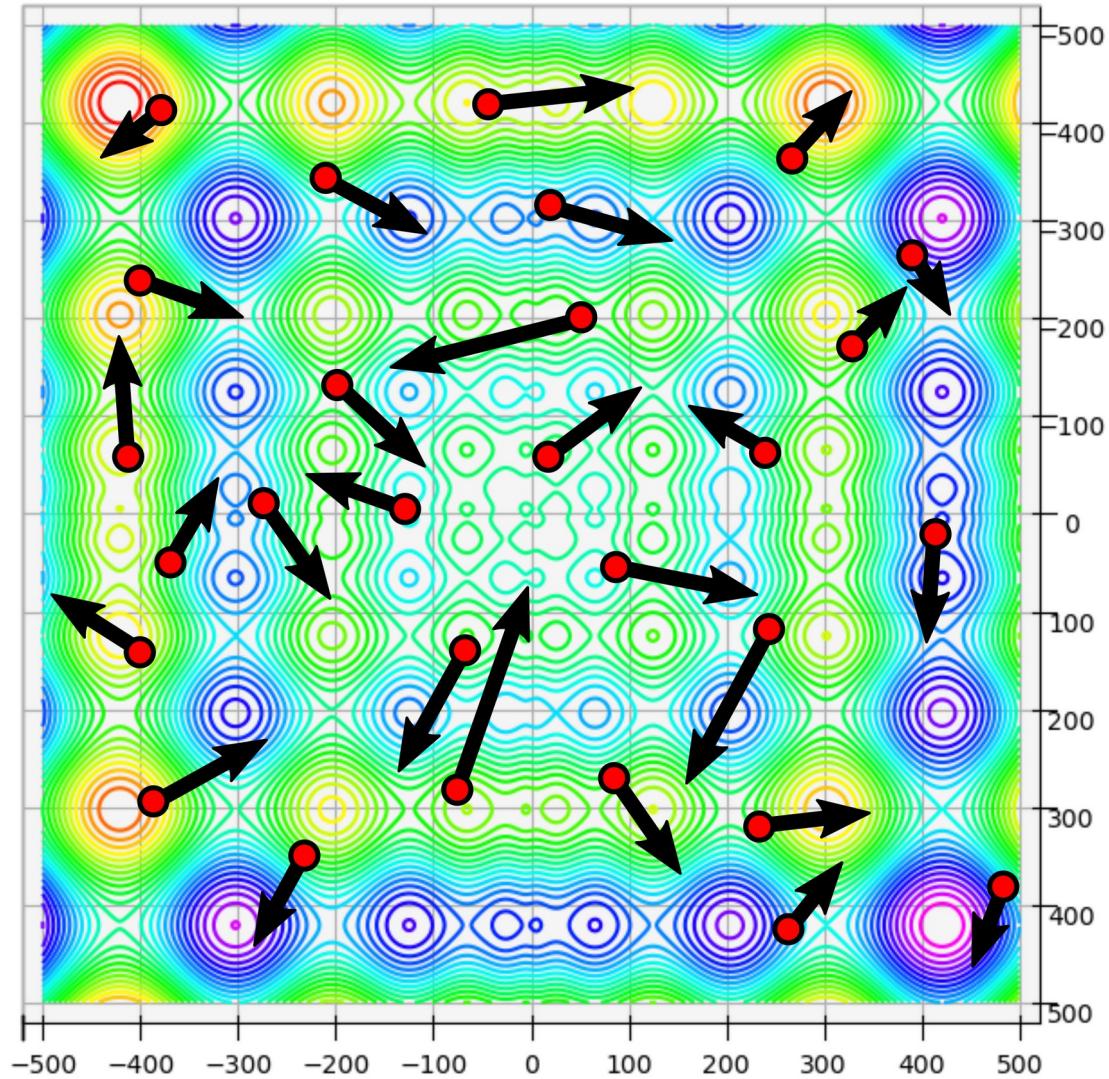
# Идея алгоритма



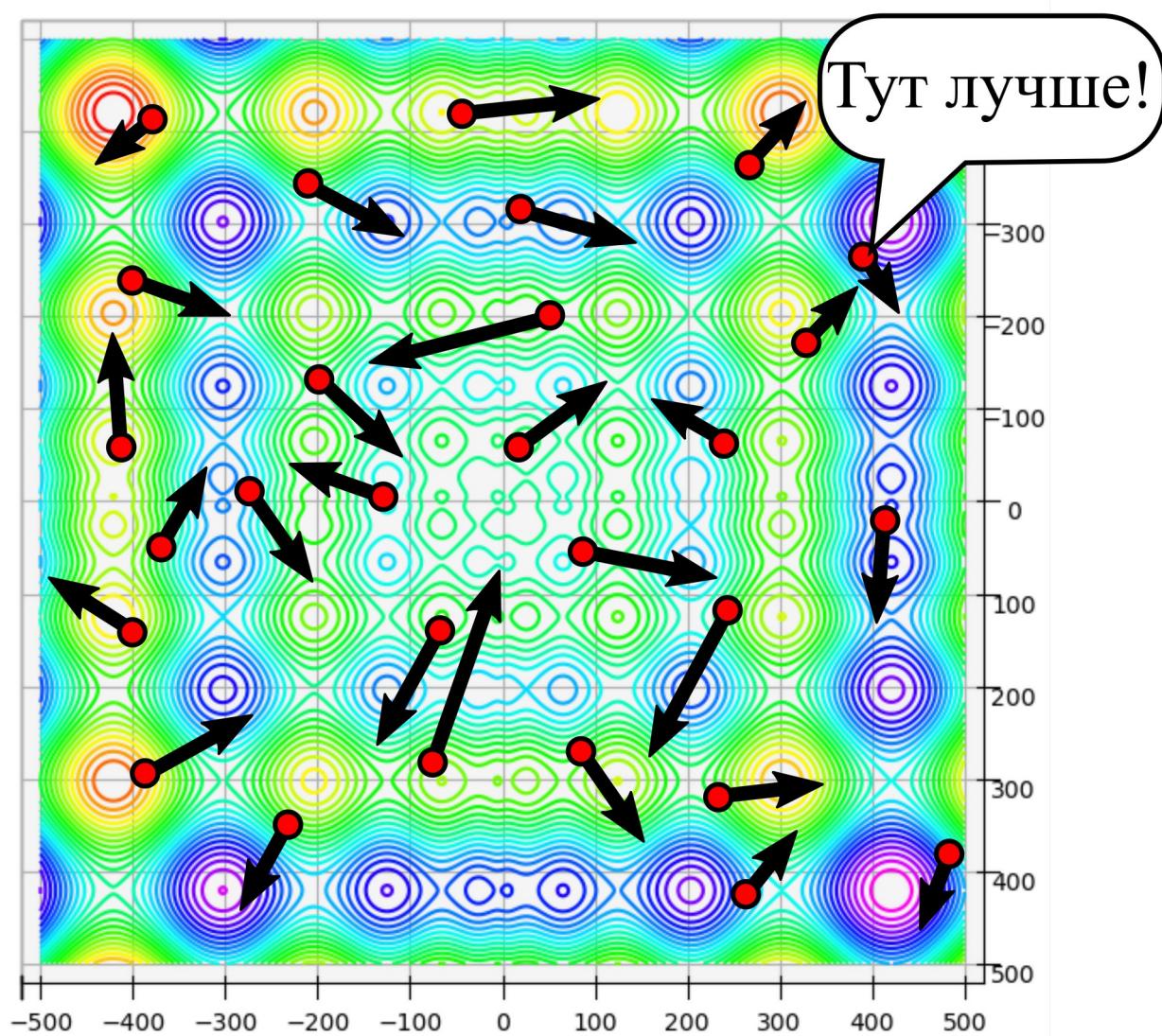
# Алгоритм роя частиц



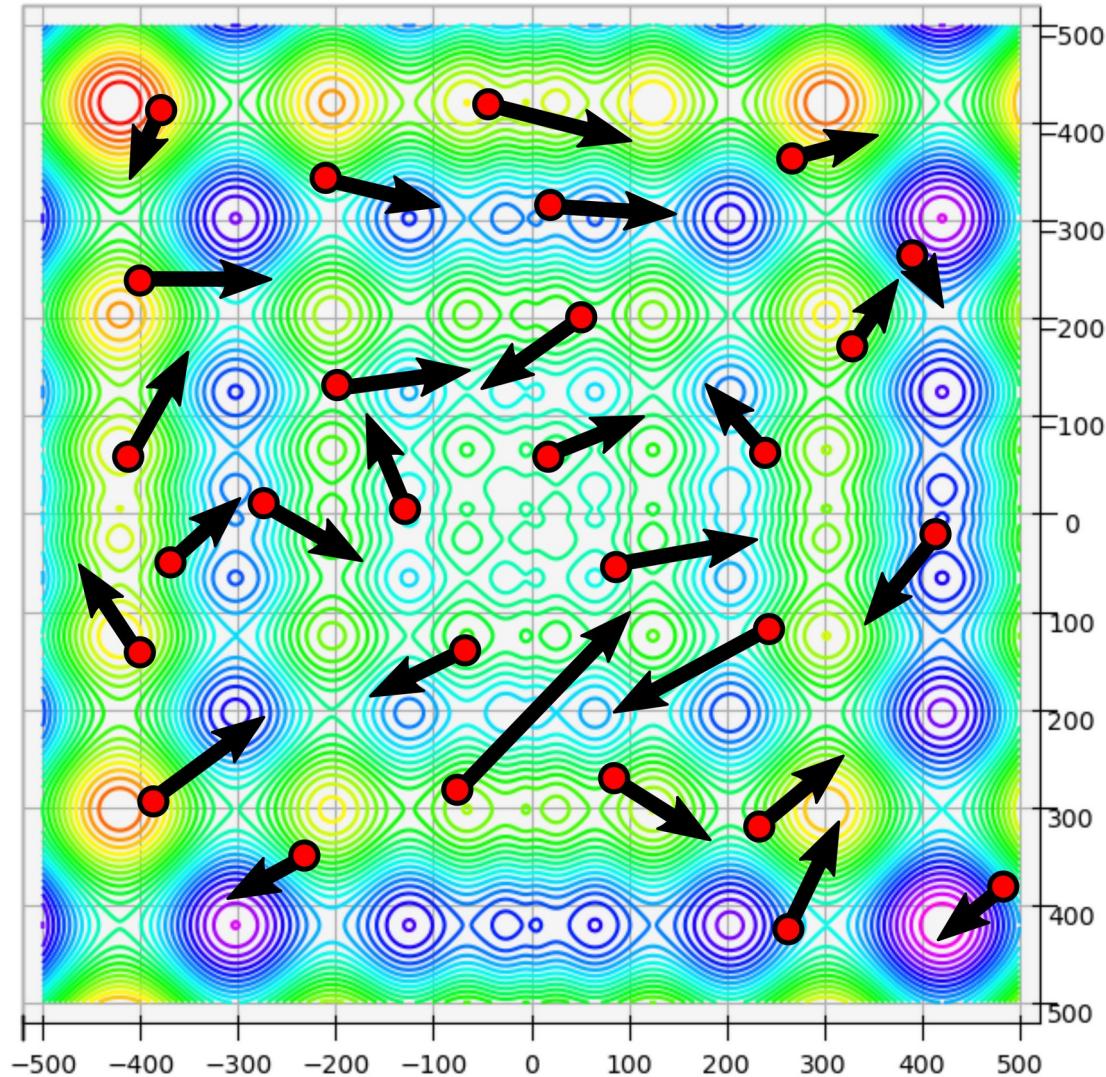
# Алгоритм роя частиц



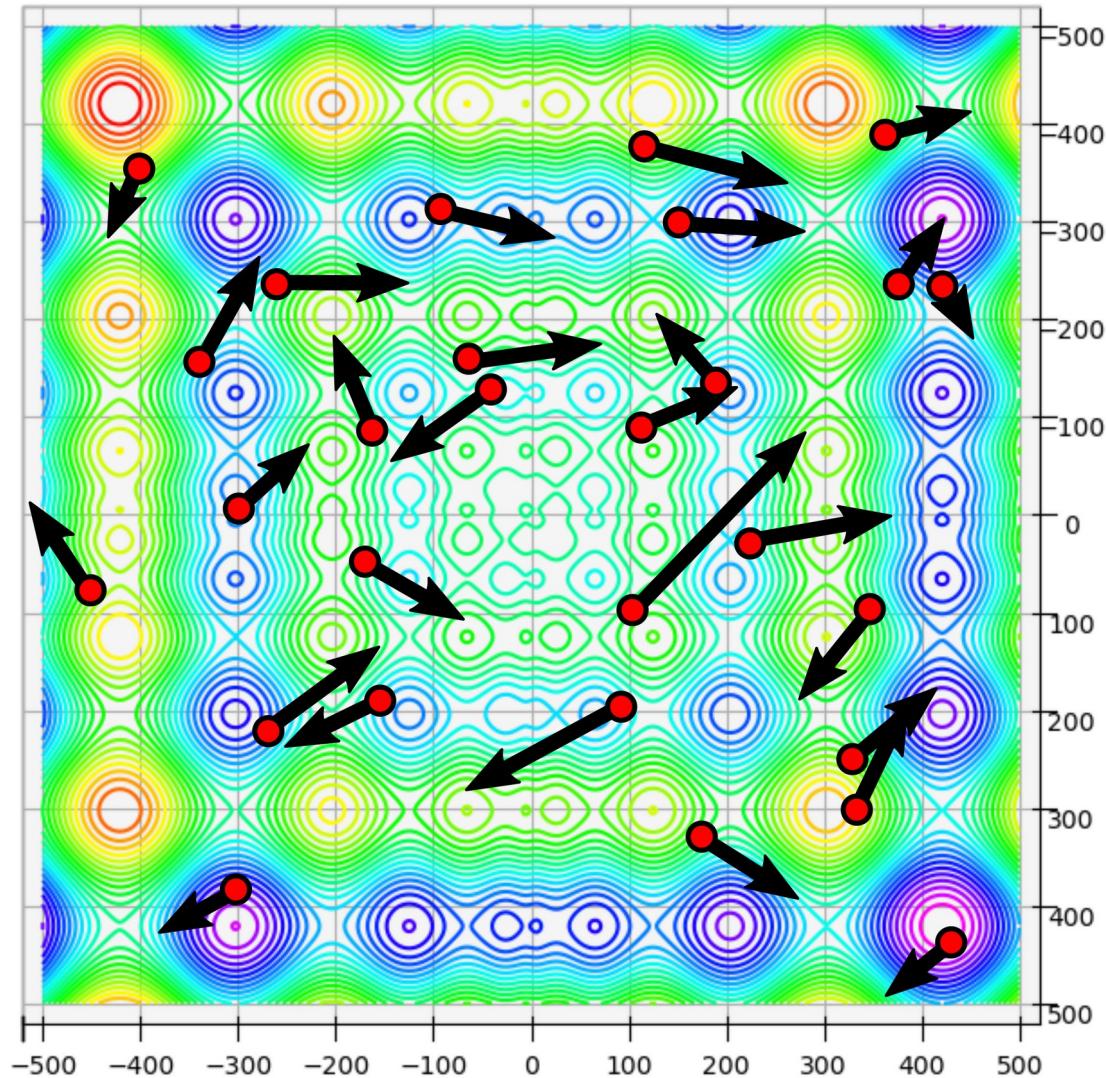
# Алгоритм роя частиц



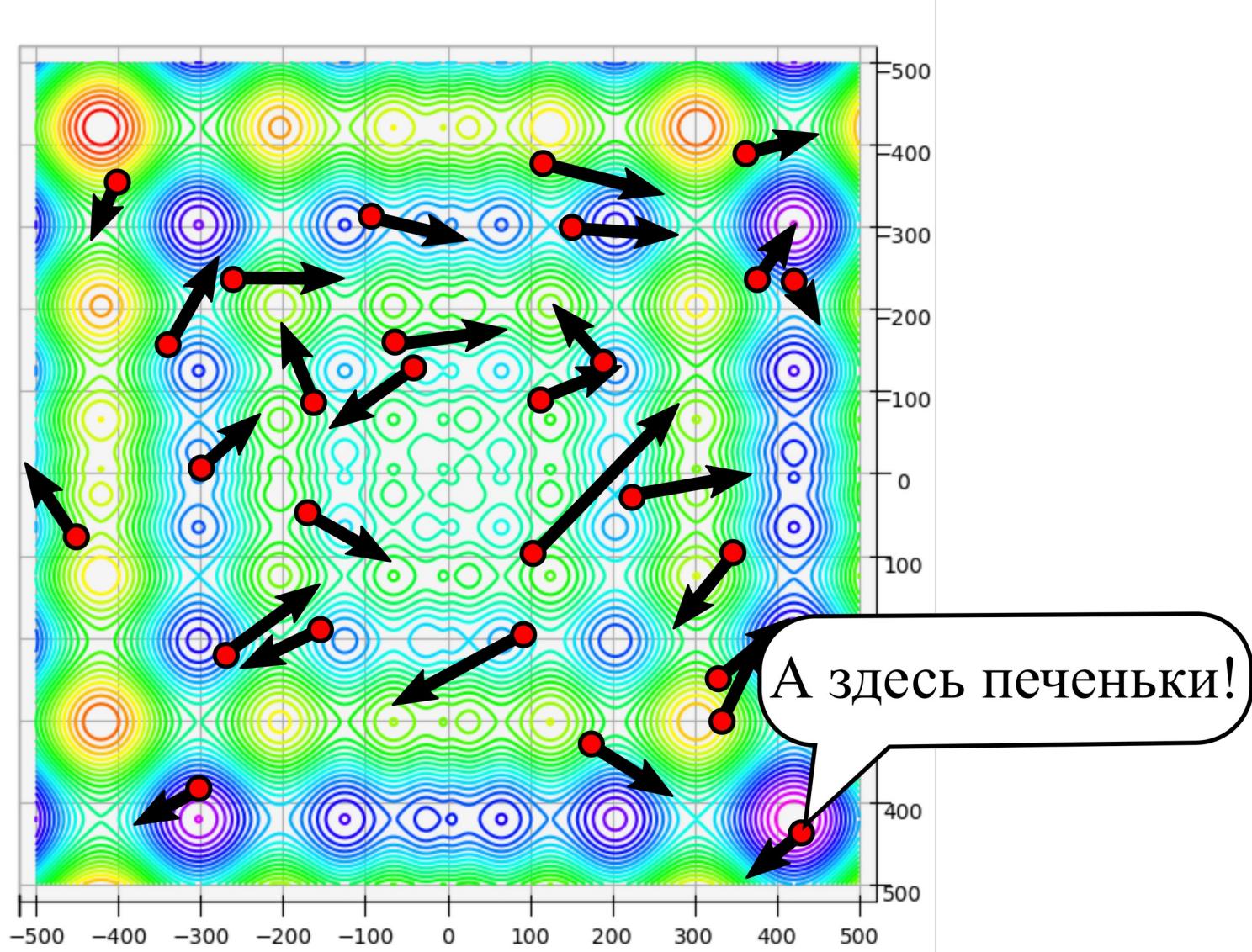
# Алгоритм роя частиц



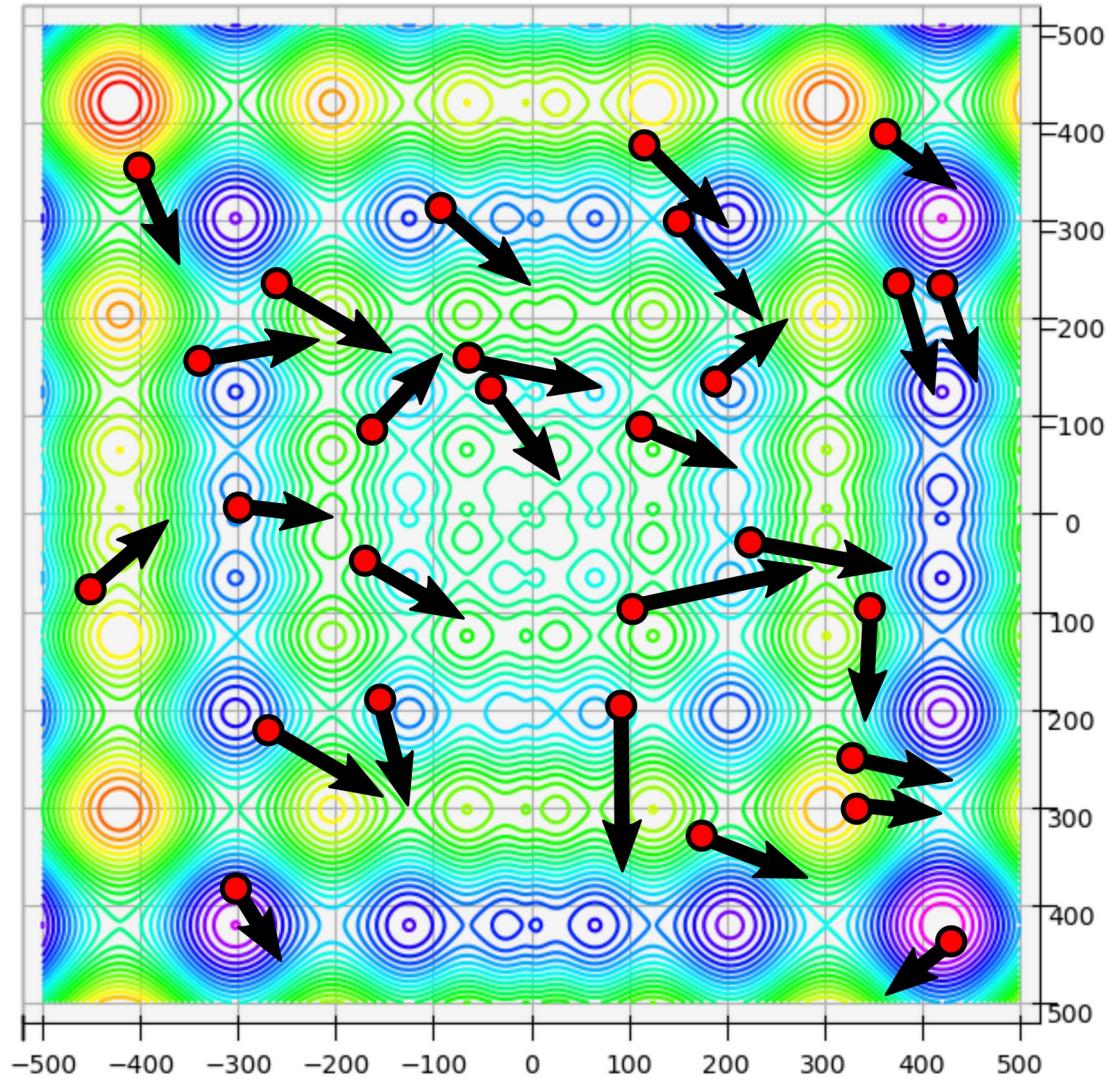
# Алгоритм роя частиц



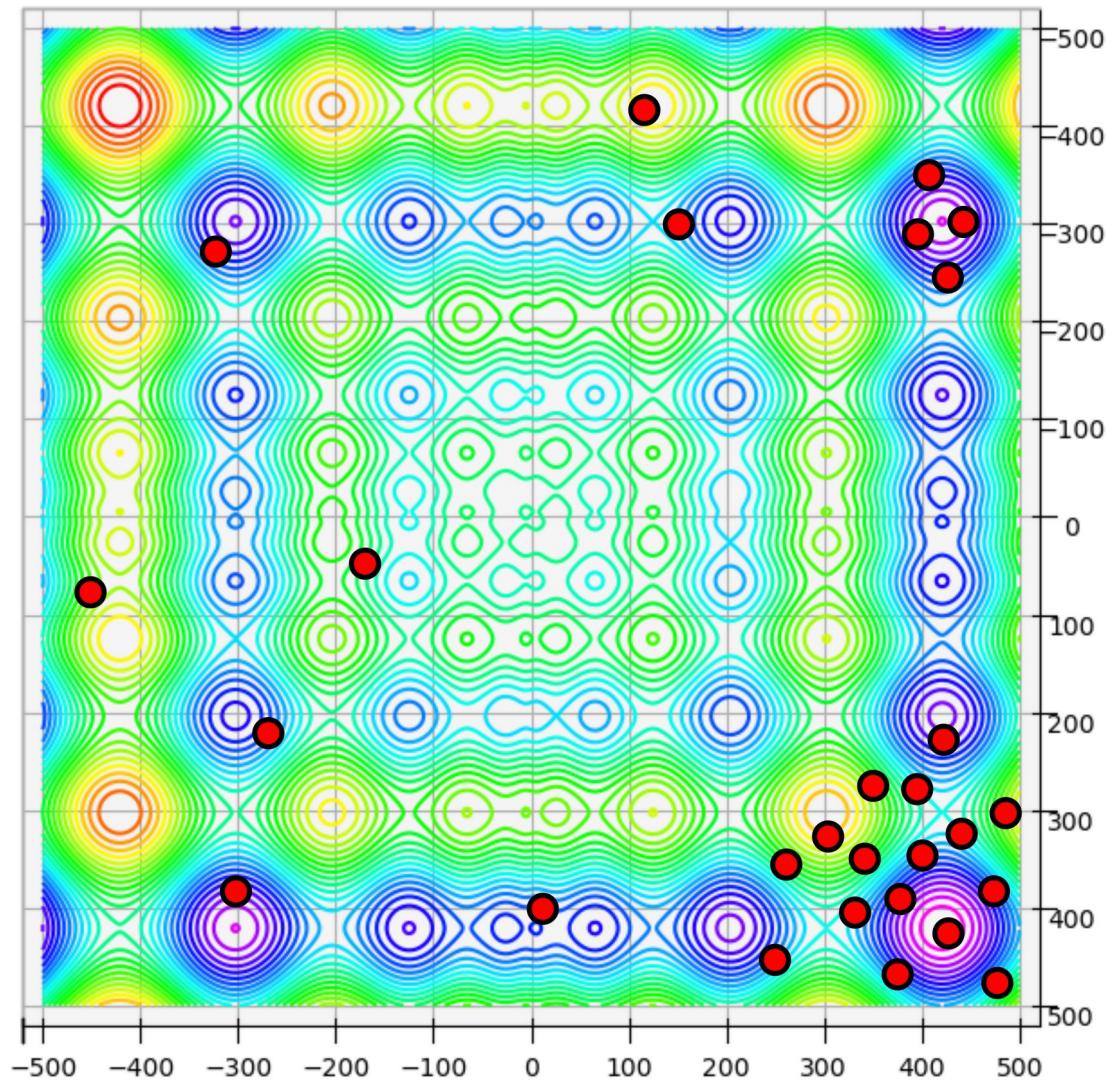
# Алгоритм роя частиц



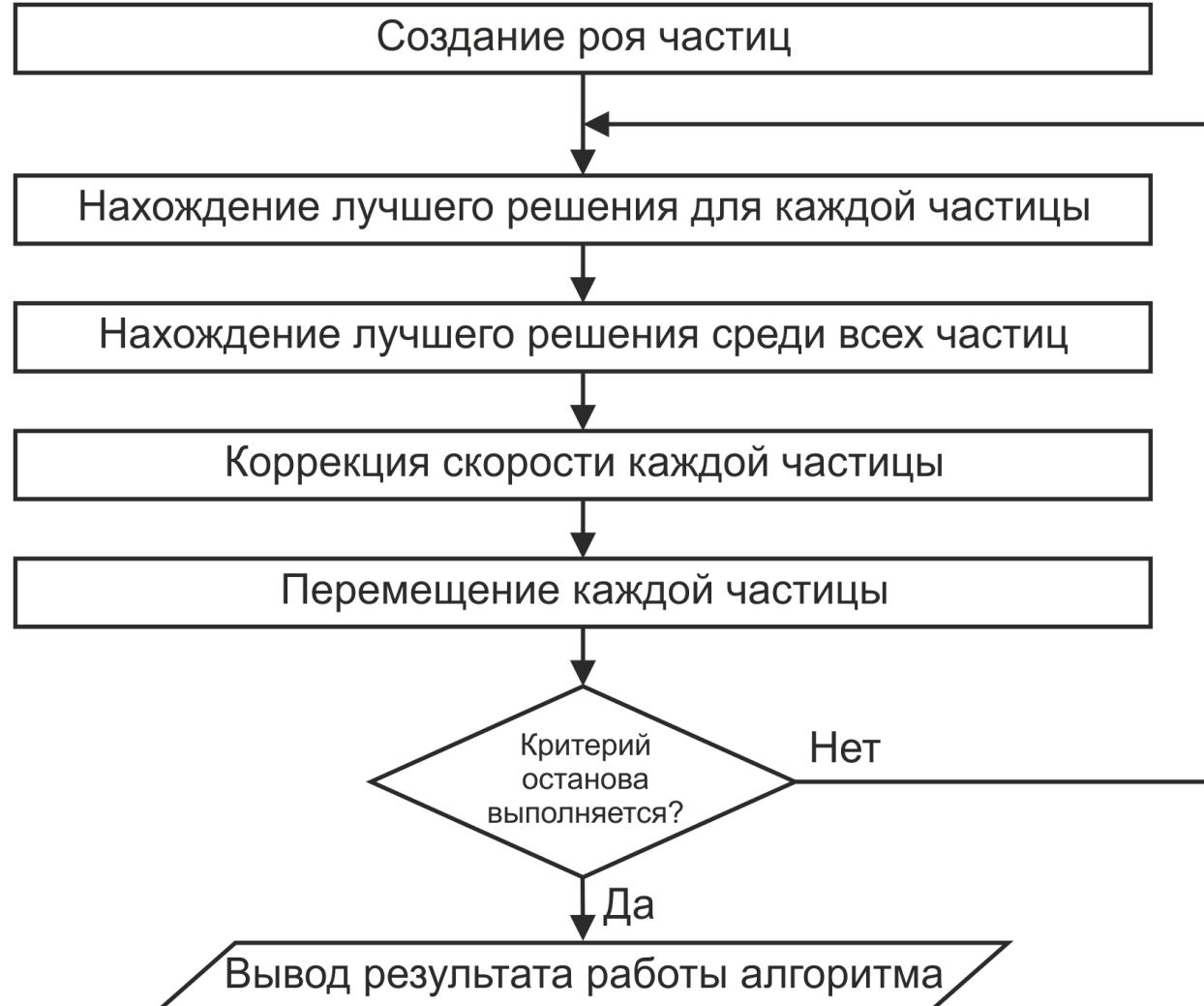
# Алгоритм роя частиц



# Алгоритм роя частиц



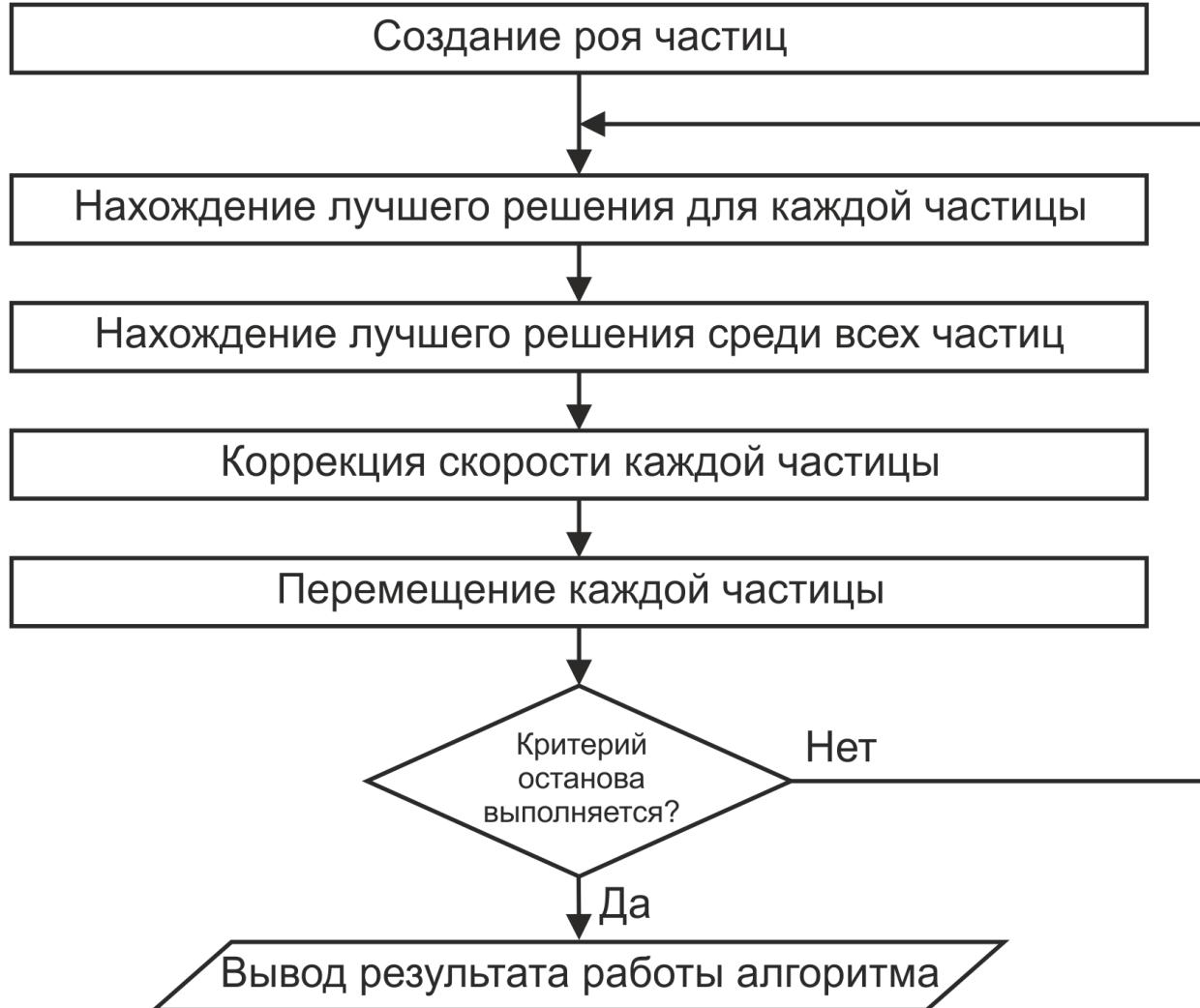
# Алгоритм роя частиц



## Критерии останова

- Постоянство целевой функции.
- Достижение заданного значения целевой функции.
- Достижение определенного номера итерации.
- Все частицы сошлись в одну точку.

# Алгоритм роя частиц



# Расчет скорости частиц.

## Классический алгоритм

$$\mathbf{v}_{i,t+1} = \mathbf{v}_{i,t} + \underbrace{\varphi_p r_p (\mathbf{p}_{i,t} - \mathbf{x}_{i,t})}_{\Delta v_{собств}} + \underbrace{\varphi_g r_g (\mathbf{g}_t - \mathbf{x}_{i,t})}_{\Delta v_{глоб}}$$

$\mathbf{v}_{i,t}$  – вектор скорости  $i$ -й частицы при  $t$ -й итерации алгоритма.

$\mathbf{x}_{i,t}$  – вектор координат  $i$ -й частицы при  $t$ -й итерации алгоритма.

$\mathbf{p}_i$  – вектор координат лучшего решения, найденного  $i$ -й частицей.

$\mathbf{g}$  – вектор координат лучшего решения, найденного всеми частицами при  $t$ -й итерации алгоритма.

$r_p, r_g$  – случайные числа в интервале  $(0, 1)$ .

$\varphi_p, \varphi_g$  – весовые коэффициенты.

# Модификации алгоритма роя частиц. Реализация «LBEST»

$$\mathbf{v}_{i,t+1} = \mathbf{v}_t + \varphi_p r_p (\mathbf{p}_{i,t} - \mathbf{x}_{i,t}) + \varphi_l r_l (\underline{\mathbf{l}}_{i,t} - \mathbf{x}_{i,t})$$

$\mathbf{l}_t$  – вектор координат лучшего решения, найденного среди соседних частиц.

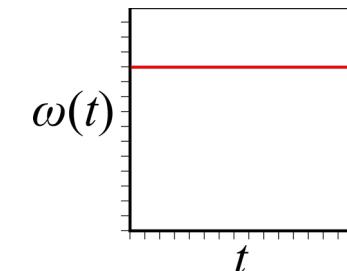
# Модификации алгоритма роя частиц. Реализация с учетом инерции

$$\mathbf{v}_{i,t+1} = \underline{\omega(t)} \mathbf{v}_{i,t} + \varphi_p r_p (\mathbf{p}_{i,t} - \mathbf{x}_{i,t}) + \varphi_g r_g (\mathbf{g}_t - \mathbf{x}_{i,t})$$

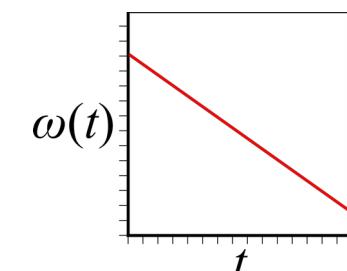
$\omega(t)$  – коэффициент инерции

# Примеры зависимостей коэффициента инерции от номера итерации

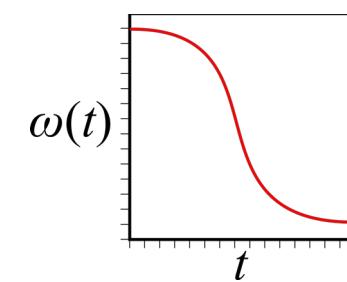
1. Постоянное значение.  $w(t) = c$



2. Линейное убывание.  $w(t) = w_{max} - \frac{w_{max} - w_{min}}{t_{max}} t$



3. Сигмоид.  $w(t) = \frac{w_{start} - w_{end}}{1 + e^{-u(t - n \cdot t_{max})}} + w_{end}, \quad u = 10^{\log(t_{max}) - 2}$



Bansal, Singh, Pramod et al. **Inertia Weight Strategies in Particle Swarm Optimization**. Proceedings of the 2011 3rd World Congress on Nature and Biologically Inspired Computing, NaBIC 2011. 633-640.  
10.1109/NaBIC.2011.6089659.

# Алгоритм роя частиц с использованием коэффициента сужения

$$\mathbf{v}_{i,t+1} = \underline{K} \left( \mathbf{v}_{i,t} + \varphi_p r_p (\mathbf{p}_{i,t} - \mathbf{x}_{i,t}) + \varphi_g r_g (\mathbf{g}_t - \mathbf{x}_{i,t}) \right)$$

$$K = \frac{2\alpha}{\varphi_p + \varphi_g - 2}$$

$$\varphi_p + \varphi_g > 4$$

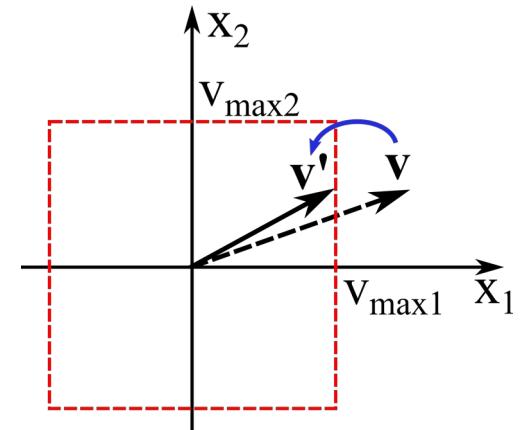
$$\alpha \in (0, 1), \text{ обычно } \alpha \approx 0.9$$

# Способы ограничения скорости частиц

1. Ограничение проекций скорости по каждой координате.

**if**  $v_{ik} > v_{\max k}$  **then**

$$v_{ik} = v_{\max k}$$



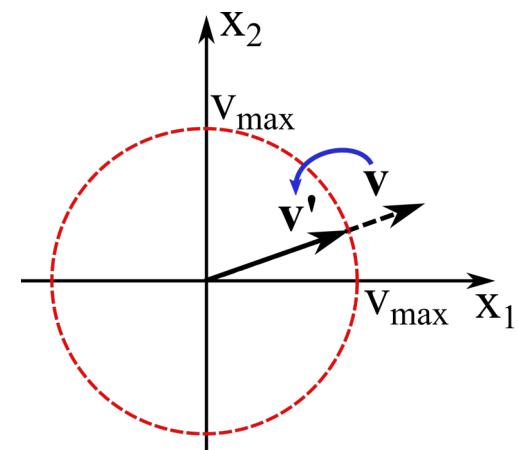
2. Ограничение модуля скорости.

**if**  $|v_i| > v_{\max}$  **then**

$$v_i = v_i \cdot v_{\max} / |v_i|$$

i — номер частицы;

k — номер размерности.



## Перемещение частиц

$$\mathbf{X}_{i,t+1} = \mathbf{X}_{i,t} + \mathbf{V}_{i,t+1}$$

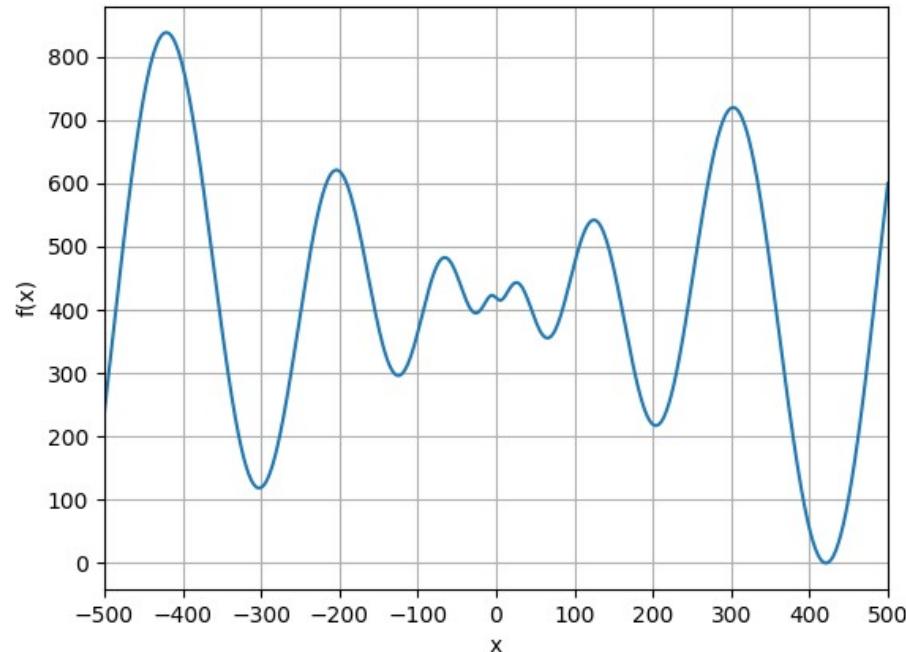
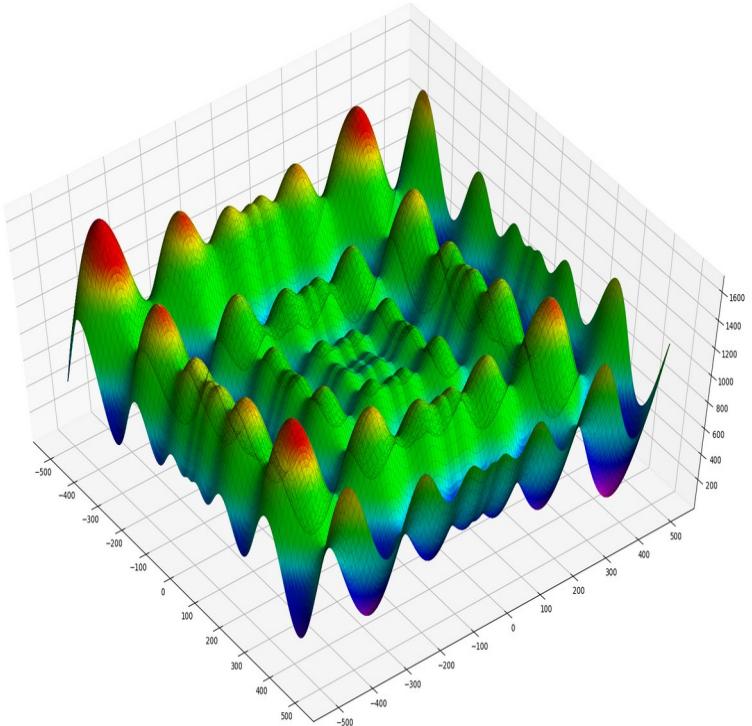
---

# Примеры работы алгоритма роя частиц

# Тестовая функция. Функция Швефеля (Schwefel function)

$$f(\mathbf{x}) = 418.9829n + \sum_{i=1}^n \left( -x_i \sin \left( \sqrt{|x_i|} \right) \right)$$

Глобальный минимум:  $f(\mathbf{x}) = 0$  при  
 $x_i = 420.9687, i = 1, \dots, n; -500 \leq x_i \leq 500$

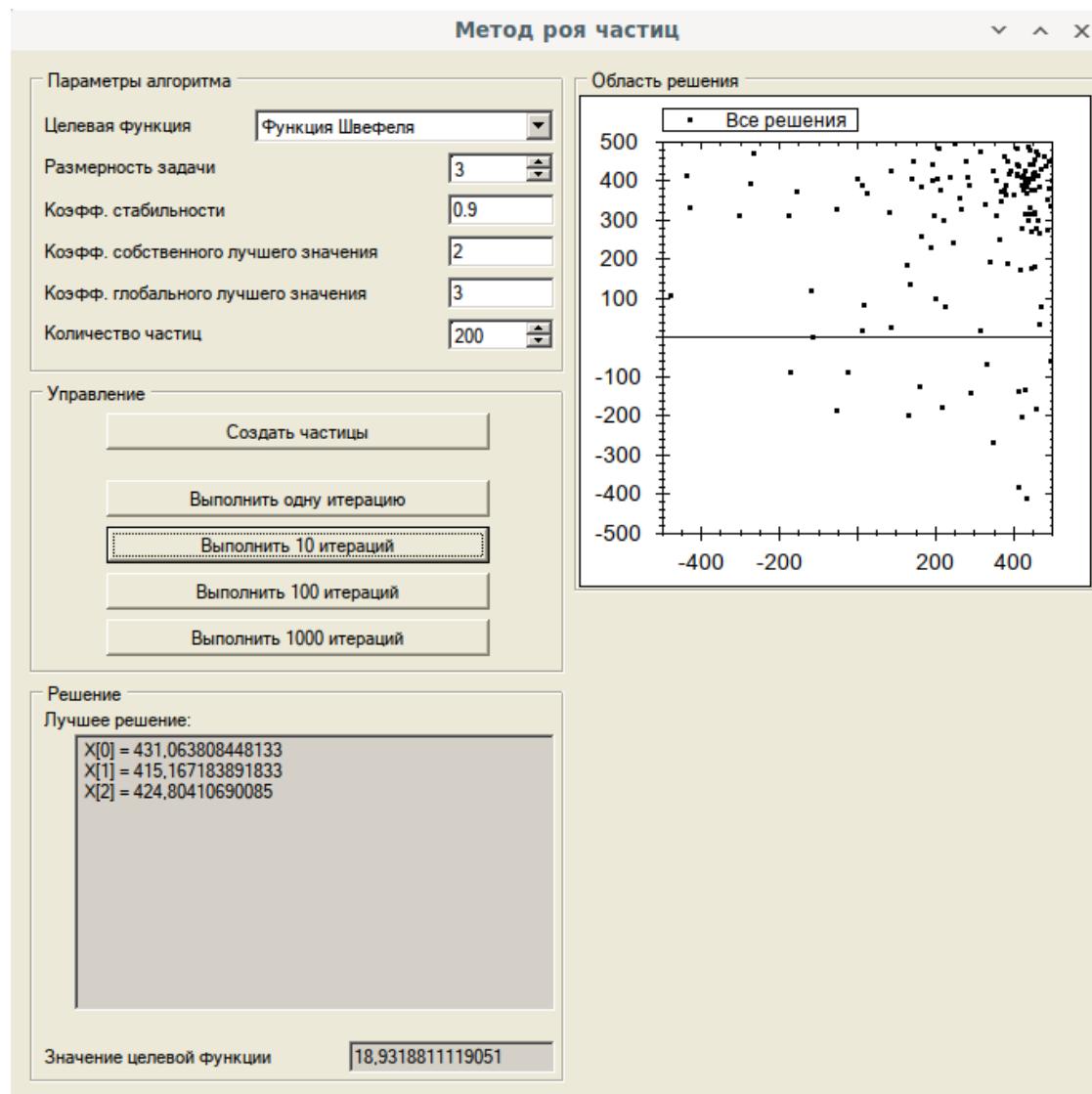


# Реализация алгоритма роя частиц на языке C#

<https://jenyay.net/Programming/ParticleSwarm>

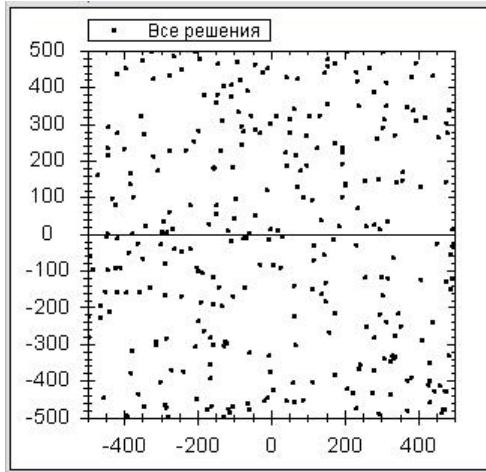


# Демонстрация работы алгоритма роя частиц

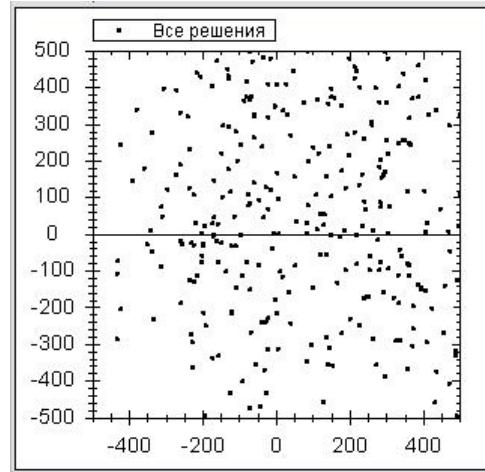


# Сходимость алгоритма роя частиц

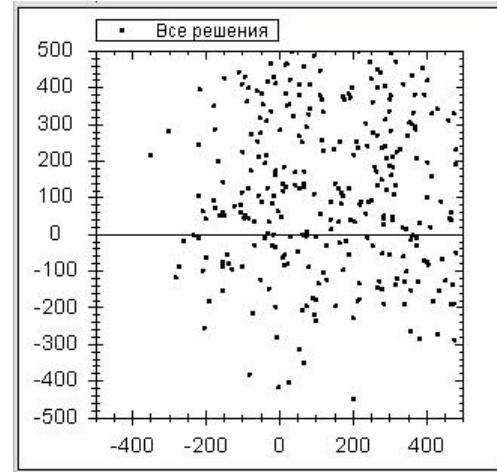
Итерация №0



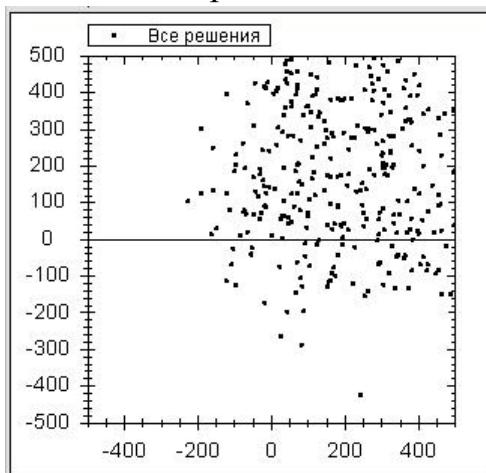
Итерация №1



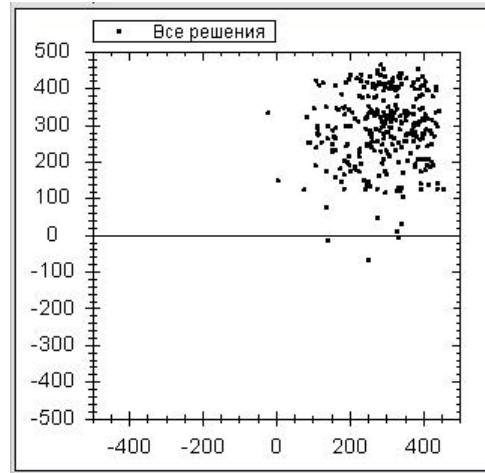
Итерация №2



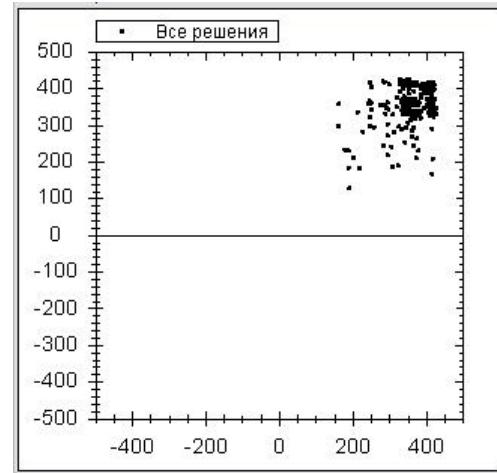
Итерация №3



Итерация №10



Итерация №30



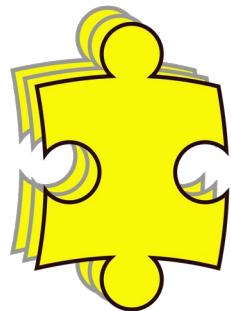
# Реализация алгоритма роя частиц на языке Rust (библиотека optlib)

<https://github.com/jenyay/rust-optimization>

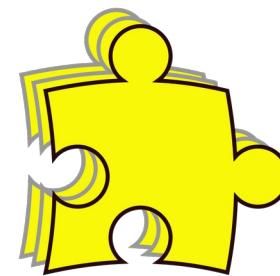


# Структура библиотеки optlib

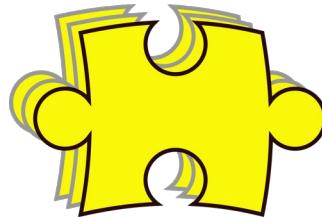
Goal



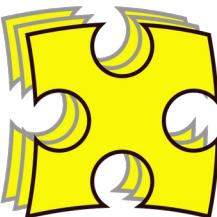
StopChecker



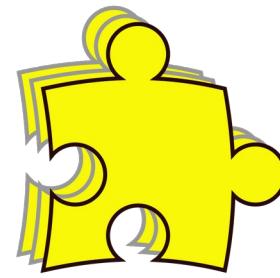
Statistics



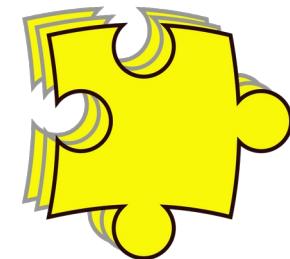
Logger



VelocityCalculator



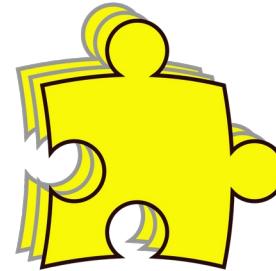
Inertia



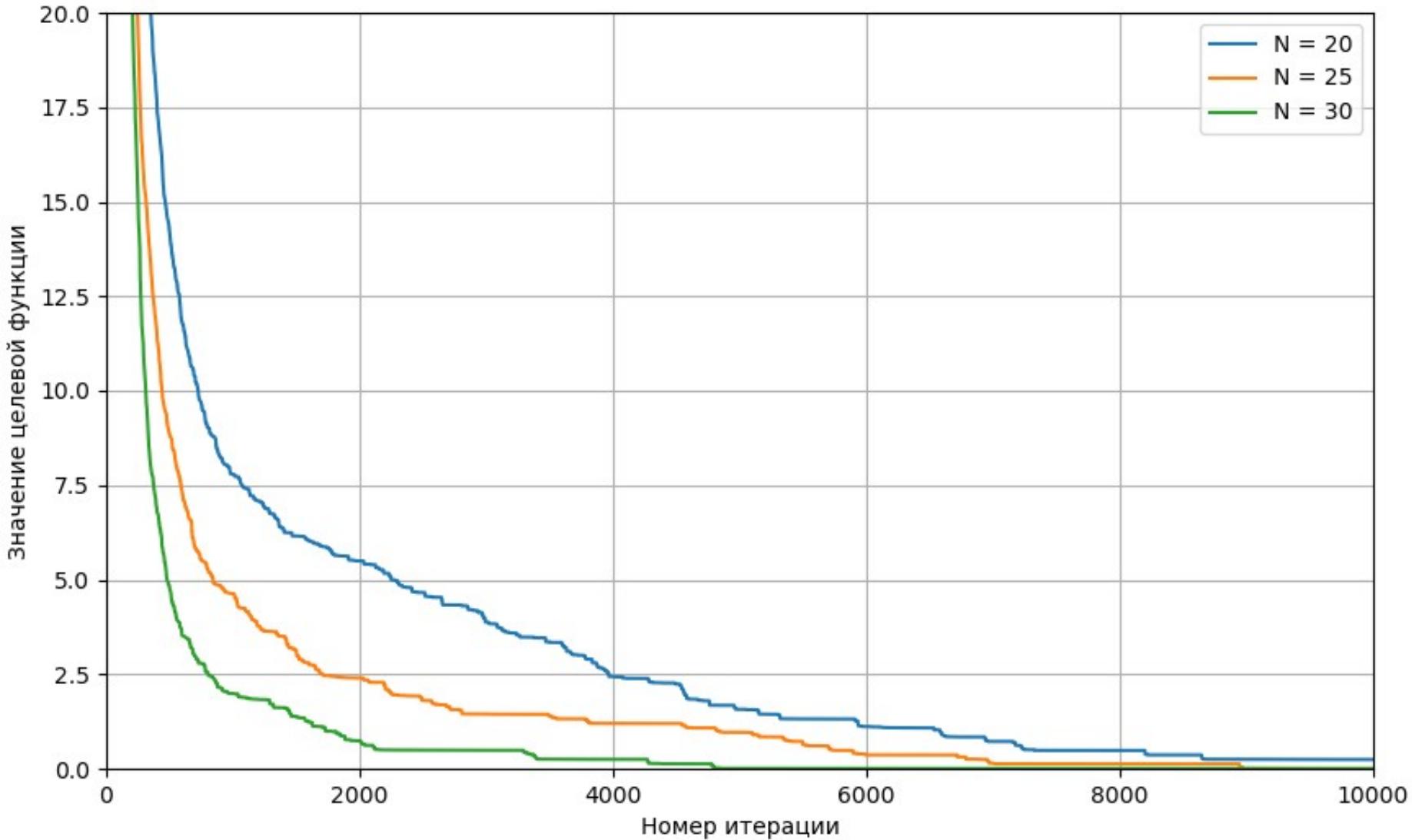
PostVelocity



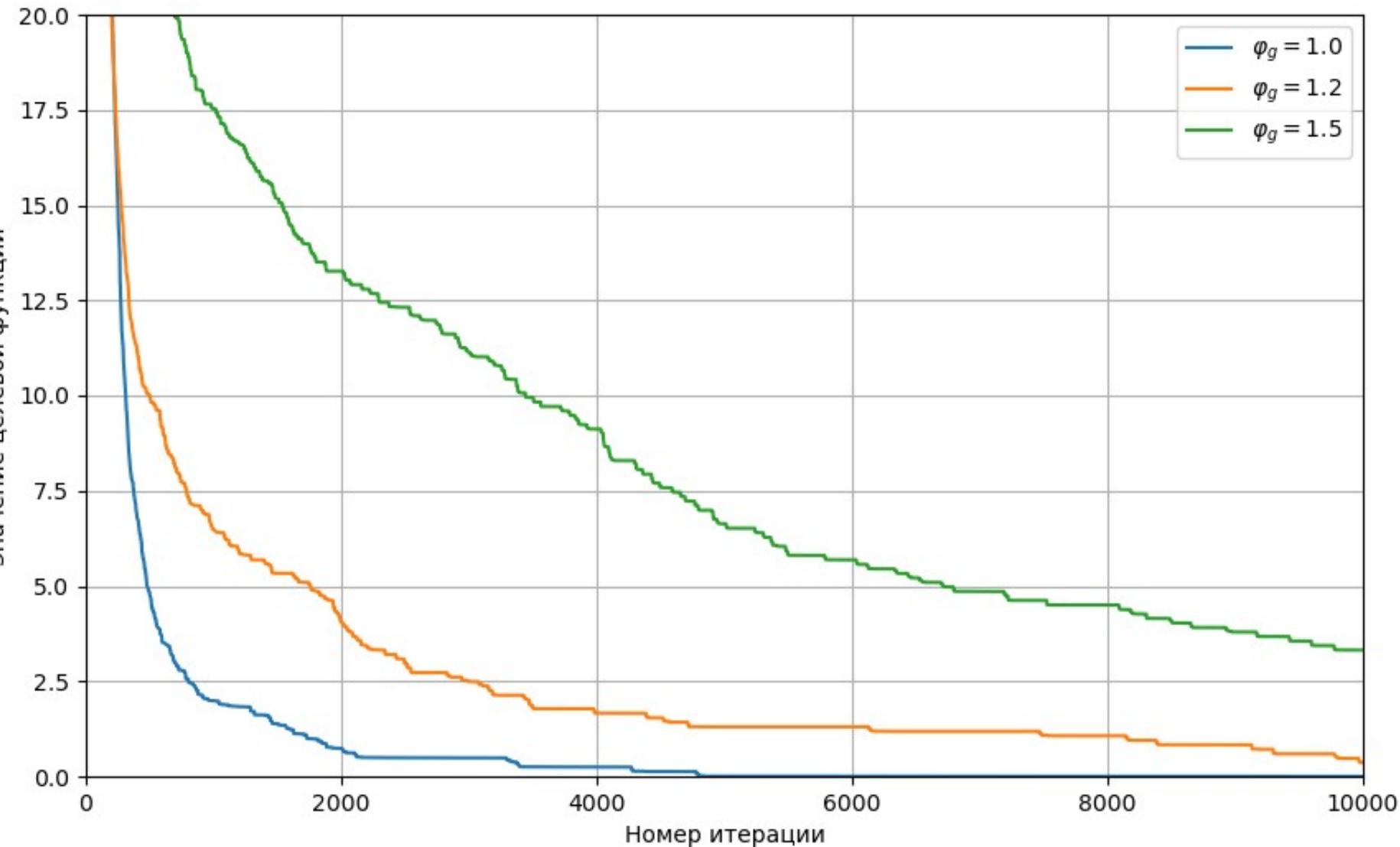
PostMove



# Влияние количества частиц на сходимость



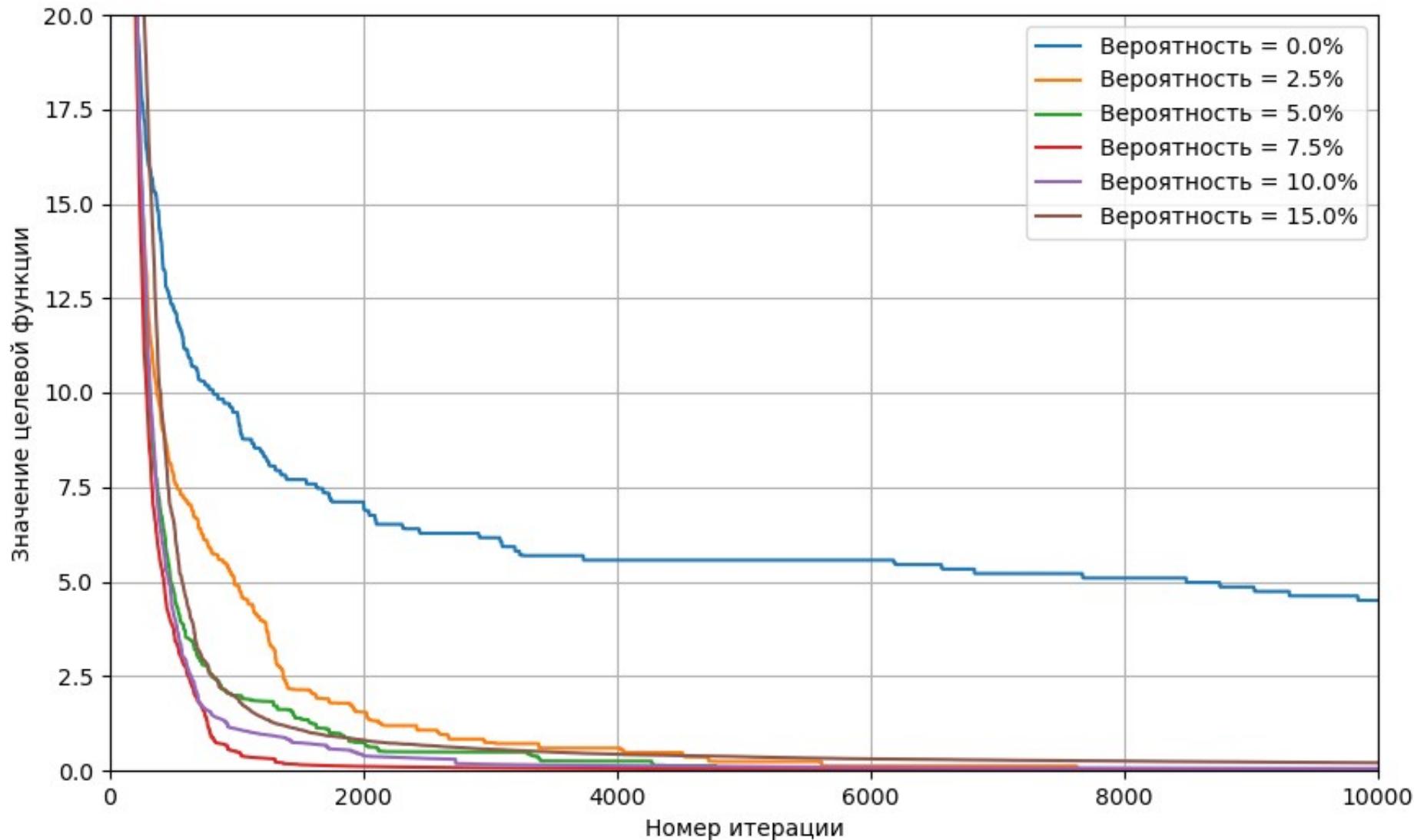
# Влияние параметра $\varphi_g$ на сходимость



# Модификации метода роя частиц

1. Добавление случайной телепортации (мутации).
  2. Алгоритм с отрицательным подкреплением.
  3. Совместное использование с другими алгоритмами оптимизации [1].
  4. Разделение роя частиц на несколько групп.
- 
1. Saptarshi Sengupta, Sanchita Basak, Richard II. (2018). **Particle Swarm Optimization: A survey of historical and recent developments with hybridization perspectives.** 10.3390/make1010010.

# Влияние вероятности случайной телепортации на сходимость



# Модификации метода роя частиц

1. Добавление случайной телепортации (мутации).
  2. Алгоритм с отрицательным подкреплением.
  3. Совместное использование с другими алгоритмами оптимизации [1].
  4. Разделение роя частиц на несколько групп.
- 
1. Saptarshi Sengupta, Sanchita Basak, Richard II. (2018). **Particle Swarm Optimization: A survey of historical and recent developments with hybridization perspectives.** 10.3390/make1010010.

# Преимущества и недостатки метода роя частиц



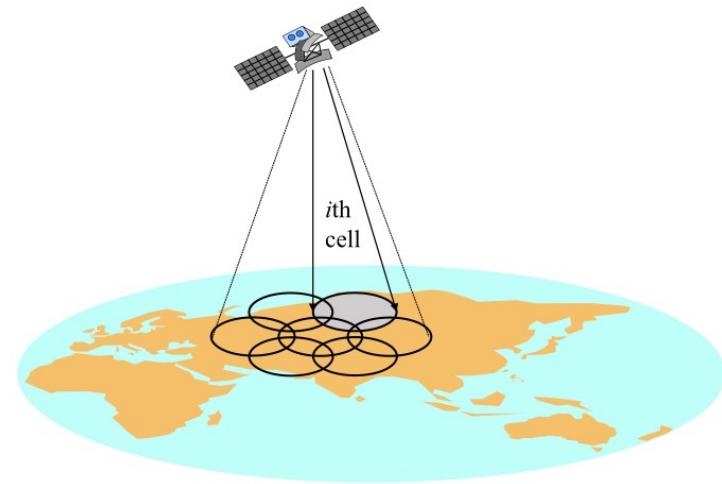
Простота реализации



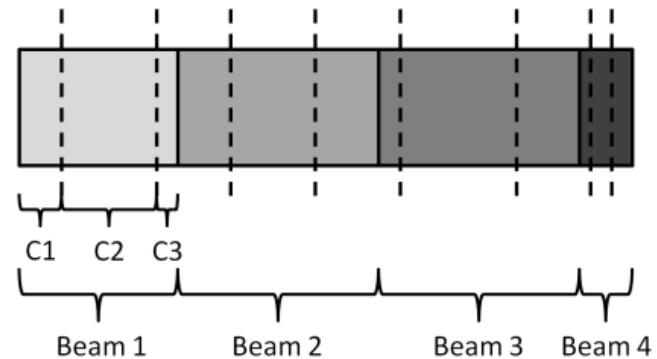
Сильная зависимость качества  
сходимости от выбранных коэффициентов

# Примеры применения в радиотехнике

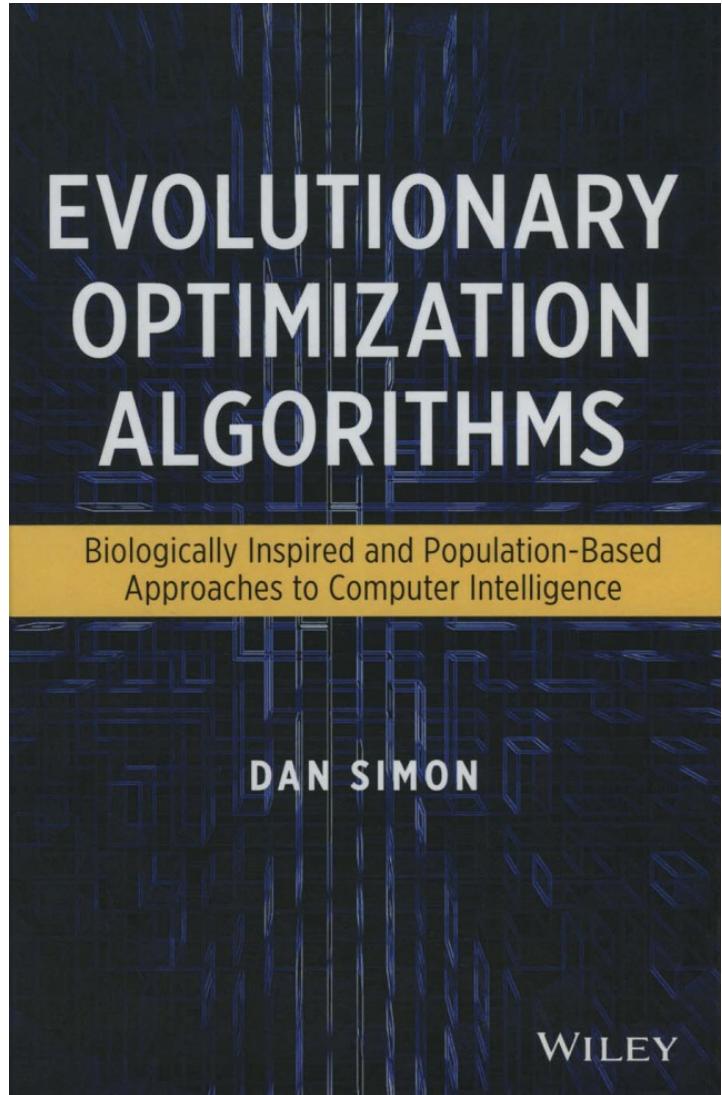
1. Fabio Renan Durand, Taufik Abrão.  
**Power allocation in multibeam satellites based on particle swarm optimization.**  
AEU - International Journal of Electronics and Communications. August 2017



2. Pachler, Nils & Garau Luis, Juan Jose & Guerster, Markus & Crawley, Edward & Cameron, Bruce. **Allocating Power and Bandwidth in Multibeam Satellite Systems using Particle Swarm Optimization.** In 2020 IEEE Aerospace Conference, 2020.



# Литература



## Ссылки

Алгоритм роя частиц.

Описание и реализации на языках Python и C#:

<https://jenyay.net/Programming/ParticleSwarm>

Реализация алгоритмов оптимизации на языке Rust:

<https://github.com/Jenyay/rust-optimization>

# Конспект доклада

- Что такое оптимизация.
- Алгоритм роя частиц.
- Способы расчета скорости частиц.
- Расчет скорости частиц с коэффициентом инерции.
- Способы ограничения скорости частиц.
- Реализация на языке C# (приложение с GUI).
- Реализация на языке Rust.
- Графики сходимости алгоритма роя частиц.
- Преимущества и недостатки алгоритма.
- Примеры применения алгоритма в радиотехнике.
- Ссылки и литература.