

Metaheurystyki optymalizacyjne

MATEUSZ GRZYB

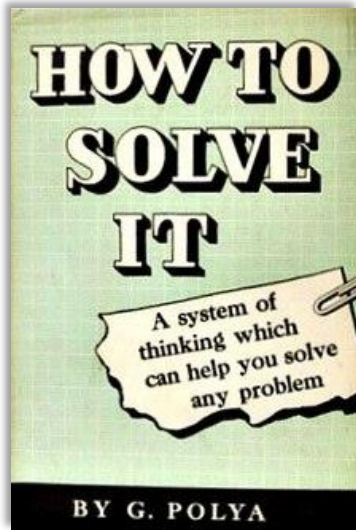
Heurystyka i metaheurystyka

- **Heurystyka** – sposób rozwiązywania problemu oparty na intuicji, przeczuciu i obserwacji zagadnienia, który nie jest poparty matematycznym dowodem otrzymania optymalnego rozwiązania.
- **Metaheurystyka** – sposób rozwiązywania dużej klasy problemów, heurystyka niewykorzystująca cech konkretnego zadania.

Metod tych używa się np. wtedy, gdy dokładny algorytm jest zbyt kosztowny lub nieznany. Najczęściej znajdowane są rozwiązania przybliżone, które w pewnych przypadkach mogą stanowić jednak punkt startowy dla algorytmu dokładnego, by zmniejszyć jego czas działania.

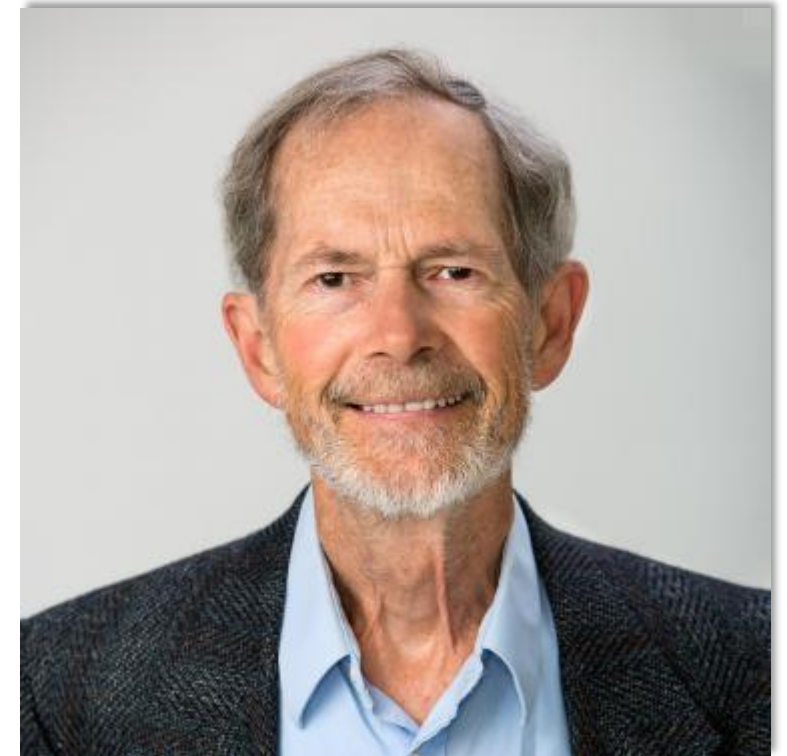


György Pólya, węgierski matematyk i autor książki „How To Solve It” (1945)



- Analogia
- Indukcja
- Analiza podproblemów

„Algorithms are conceived in analytic purity in the high citadels of academic research, heuristics are midwived by expediency in the dark corners of the practitioner’s lair. . . and are accorded lower status.” (Glover, 1977).



Fred W. Glover, twórca pojęcia „metaheurystka” i algorytmu Tabu Search (1986)

Metaheurystyki w optymalizacji

ZALETY

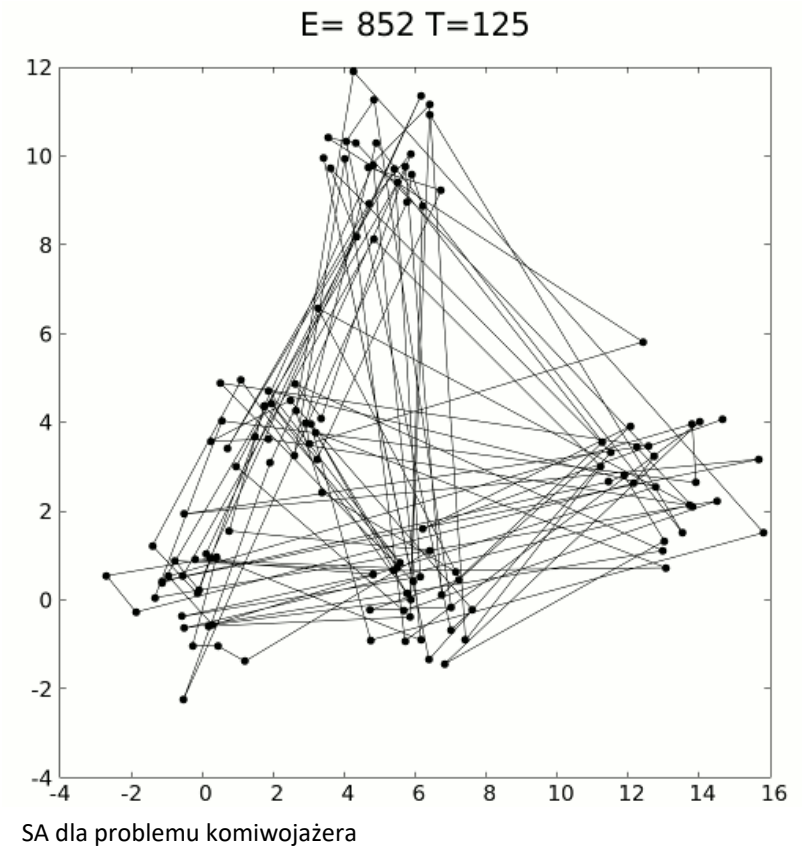
- Szybkość działania
- Mała liczba założeń
- Intuicyjne rozumienie
- Możliwości adaptacji
- Podatność na modyfikacje

WADY

- Przybliżone rozwiązania
- Brak udowodnionej zbieżności
- Niedeterministyczność wyników
- Znaczący wpływ parametrów na wyniki
- Metafory skrywające brak innowacyjności

Simulated Annealing (SA)

- Idea zaproponowana kilkukrotnie w sposób niezależny w latach 70-80,
- zastosowana do problemu komiwojażera i nazwana w **Kirkpatrick, Gelatt Jr., Vecchi (1983)**,
- inspirowana procesem wyżarzania w metalurgii,
- stosowana w wielu problemach (optymalizacja funkcji rzeczywistej, problem komiwojażera, predykcja struktury białek...),
- dowód zbieżności? **Laarhoven and Aarts (1987)**



Eksploracija V eksploatacija

RANDOM WALK

- Let $s = s_0$
- Let $s_{best} = s$
- For $k = 0$ through k_{max} (exclusive):
 - Pick a random neighbour, $s \leftarrow \text{neighbour}(s)$
 - If $f(s) > f(s_{best})$:
 - $s_{best} \leftarrow s$
- Output: the best state s_{best}

STOCHASTIC HILL CLIMBING

- Let $s = s_0$
- For $k = 0$ through k_{max} (exclusive):
 - Pick a random neighbour, $s_{new} \leftarrow \text{neighbour}(s)$
 - If $f(s_{new}) > f(s)$:
 - $s \leftarrow s_{new}$
- Output: the last state s

Eksploracja \wedge eksploatacja

SIMULATED ANNEALING

- Let $s = s_0$
- Let $s_{best} = s$
- For $k = 0$ through k_{max} (exclusive):
 - $T \leftarrow \text{temperature}(1 - (k+1) / k_{max})$
 - Pick a random neighbour, $s_{new} \leftarrow \text{neighbour}(s)$
 - If $f(s) > f(s_{best})$:
 - $s_{best} \leftarrow s$
 - If $P(E(s), E(s_{new}), T) \geq \text{random}(0, 1)$:
 - $s \leftarrow s_{new}$
- Output: the best state s_{best}



SA dla jednowymiarowej funkcji rzeczywistej

$T \rightarrow 0 \Rightarrow \text{if } E(s_{new}) < E(s) \text{ then } P(E(s), E(s_{new}), T) \rightarrow 0 \text{ else } P(E(s), E(s_{new}), T) \rightarrow > 0$

if $E(s_{new}) > E(s)$ then $P(E(s), E(s_{new}), T) = 1$, ale niekoniecznie

$\text{zazwyczaj } E(s) - E(s_{new}) \uparrow \Rightarrow P(E(s), E(s_{new}), T) \downarrow$

Uzasadnienie nazwy SA?

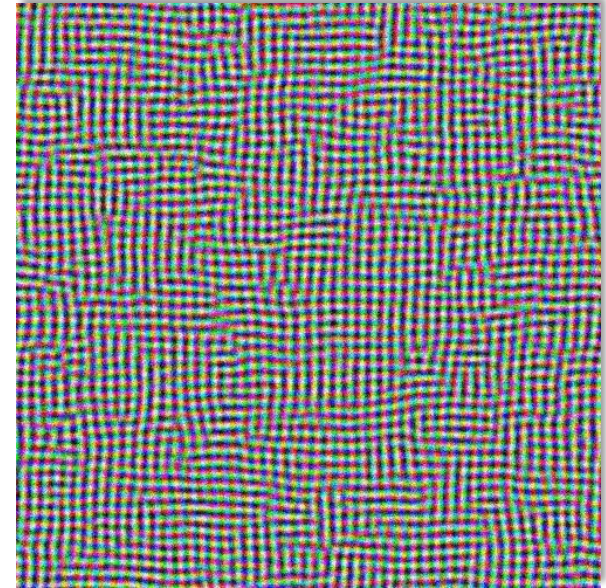
Zadanie polega na minimalizacji pewnej energii potencjalnej siły, która powoduje przyciągnięcie podobnych kolorów na małym dystansie i odpychanie na dużym dystansie.

Ruch polega na zamianie dwóch pikseli miejscami.

Efekt dla wolnego stygnięcia przypomina strukturę amorficzną, a efekt dla szybkiego stygnięcia strukturę krystaliczną.



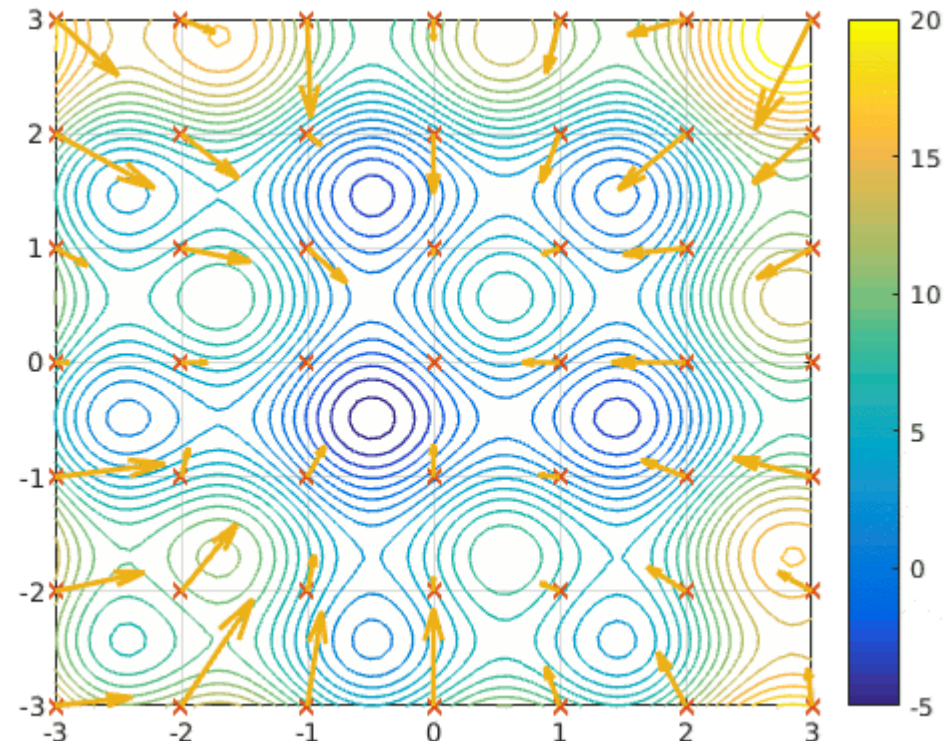
Efekt dla szybkiego stygnięcia



Efekt dla wolnego stygnięcia

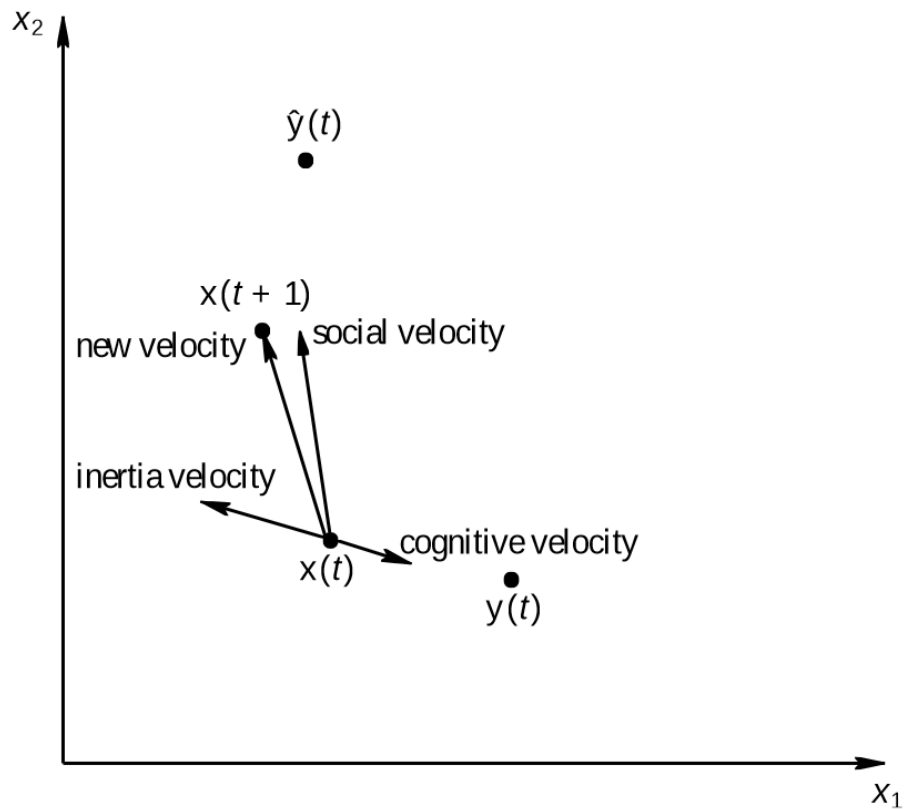
Particle Swarm Optimization (PSO)

- Algorytm zaproponowany w **Eberhart, Kennedy (1995)**
- przykład idei inteligencji rojowej (inteligentne zachowanie zespołu prostych bytów),
- początkowo zamierzony do symulacji zachowań społecznych, ale uproszczony i zastosowany do optymalizacji,
- inspirowany zachowaniem stada ptaków, ławicy ryb.



PSO dla dwuwymiarowej funkcji rzeczywistej

SPSO 2007



$$x(t+1) \leftarrow \omega v(t) + \text{random}(U[0, \varphi_1])(\hat{y}(t) - x(t)) + \text{random}(U[0, \varphi_2])(y(t) - x(t))$$

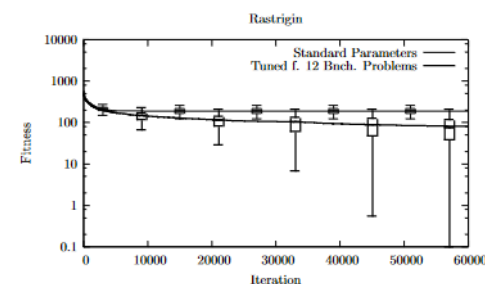
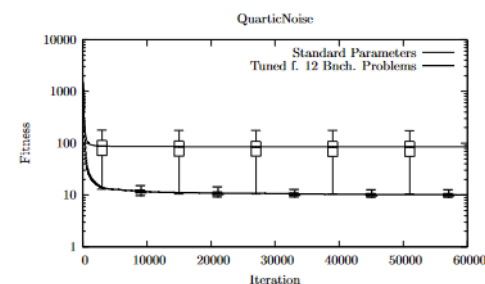
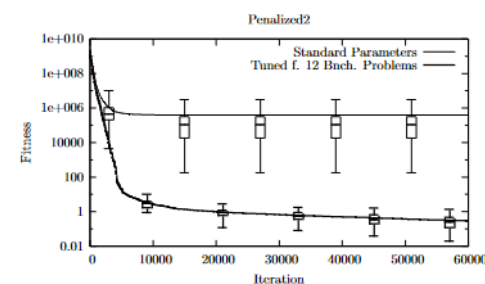
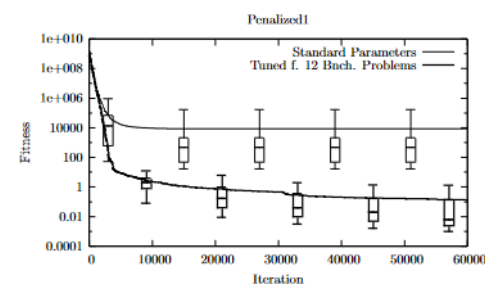
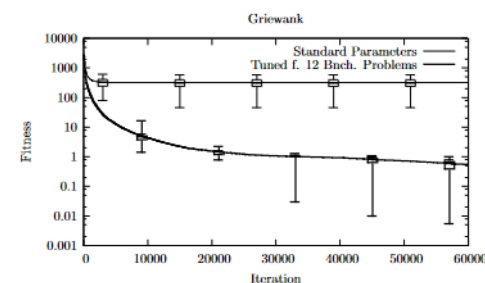
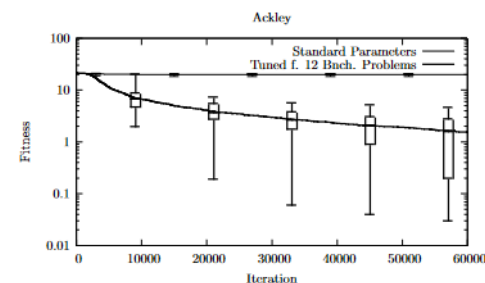
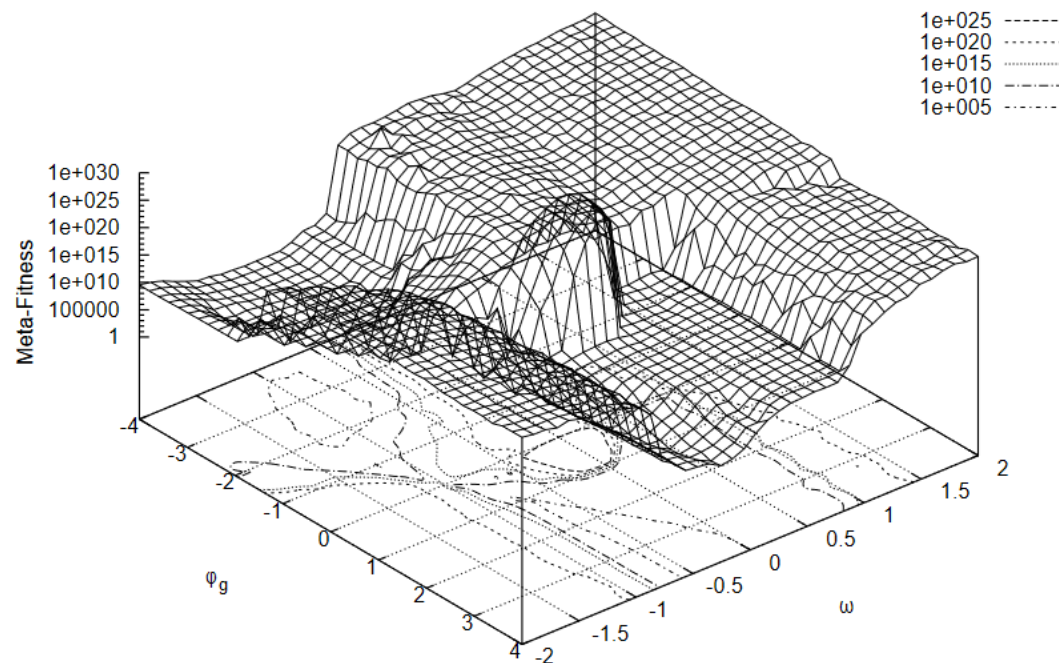
$x(t)$ – obecna pozycja
 $v(t)$ – obecna prędkość
 $v(t+1)$ – nowa prędkość
 $y(t)$ – lokalne optimum
 $\hat{y}(t)$ – globalne optimum

$\varphi_1, \varphi_2, \omega$ - parametry

Zambrano-Bigiarini, Clerc, Rojas (2011) – SPSO 2011

Wrażliwość na parametry

Wykresy pochodzą z **Pedersen (2010)**. Zauważalne są różnice kilku rzędów wielkości.



Krytyka metafory

Bees, flies, fruit flies, termites, fireflies, mayflies, glow worms...

Sörensen (2013) – „Metaheuristics - the metaphor exposed”:

- Zalew metaheurystyk inspirowanych metaforami z dziedzin biologii, socjologii, fizyki
- Podmiana usystematyzowanej terminologii na analogię
- Metafora nie jako inspiracja innowacji, a jako usprawiedliwienie jej braku
- Odwołanie do niezwiązanych zjawisk, zamiast rozważnej analizy struktury problemu
- Odwracanie uwagi od prawdziwie innowacyjnych metaheurystyk
- Brak prawidłowego pozycjonowania w utartej literaturze
- Walka o lepsze wyniki, a nie zrozumienie źródła skuteczności
- Funkcje testowe nieadekwatne do rzeczywistych problemów
- Brak ustandaryzowanej metody testowania algorytmów



Kenneth Sörensen, założyciel EURO Working Group on Metaheuristics i edytor w Journal of Metaheuristics

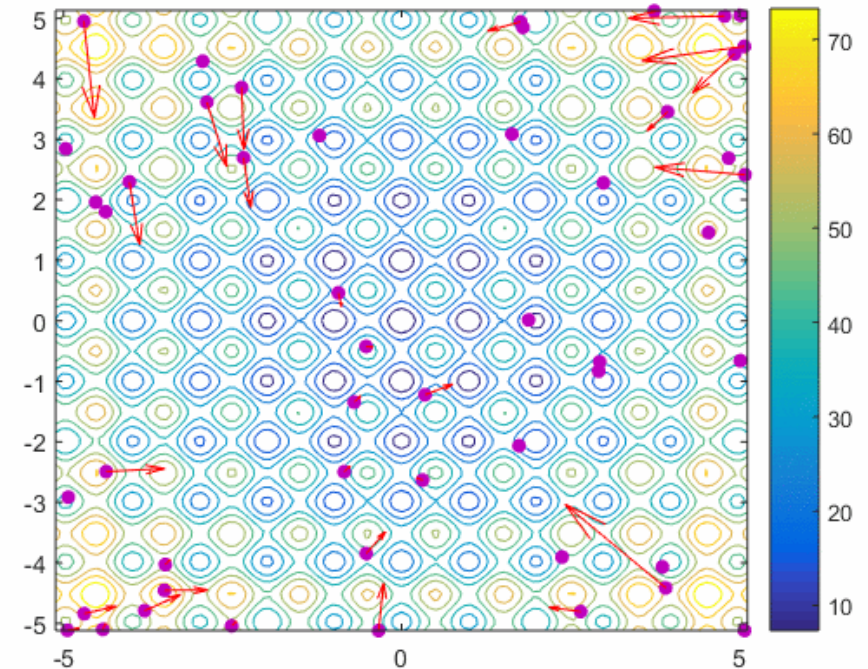
Jak opublikować 30-letnią ideę?

Geem, Kim, Loganathan (2001) – „Harmony Search”:

- Inspirowany improwizacją muzyków
- „harmony” = „solution”, „note” = „decision variable”, „sounds better” = „better objective function value”, „harmony memory” = „population”...
- **Weyland (2010)** – HS to tak naprawdę znane 30 lat wcześniej $(\mu + 1)$ -ES
- „Most importantly, when I searched Wikipedia [emphasis added], I could not find the structure $(\mu + 1)$ -ES which, the protester claimed, equals that of HS.” – Geem
- Metafora maskująca brak innowacji
- W czasach artykułu Weylanda Google Scholar zwracał ponad 500 wyników dla frazy „harmony search”, a obecnie około 1 960 000

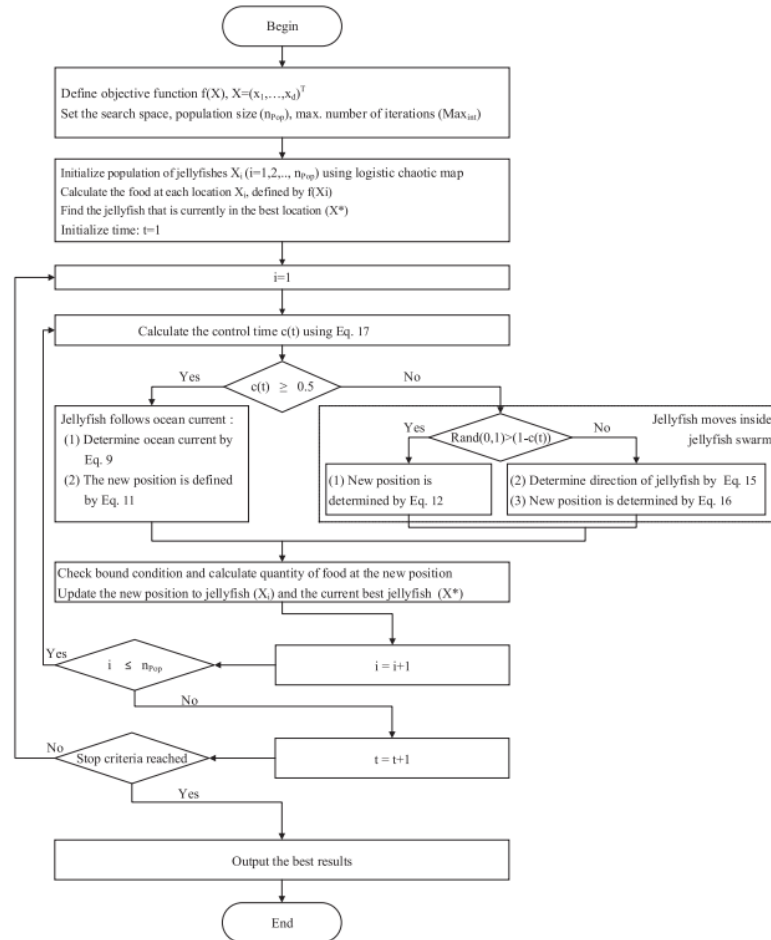
Jellyfish Search (JS)

- Algorytm zaproponowany w Chou, Truong (2020),
- opublikowany w Journal of Applied Mathematics and Computation,
- mieszane uczucia co do jakości artykułu,
- inspirowany ruchami meduz w oceanie,
- oczywiste podobieństwo do PSO, ale nie zupełnie...



JS dla dwuwymiarowej funkcji rzeczywistej

Mechanizm działania JS



Begin

For i=1: nPop do

Calculate the time control $c(t)$ using Eq. (17)

If $c(t) \geq 0.5$:

Jellyfish follows ocean current

Else: Jellyfish moves inside swarm

If $\text{rand}(0,1) > (1-c(t))$:

Jellyfish exhibits type A motion (Passive motions)

Else:

Jellyfish exhibits type B motion (Active motions)

End if

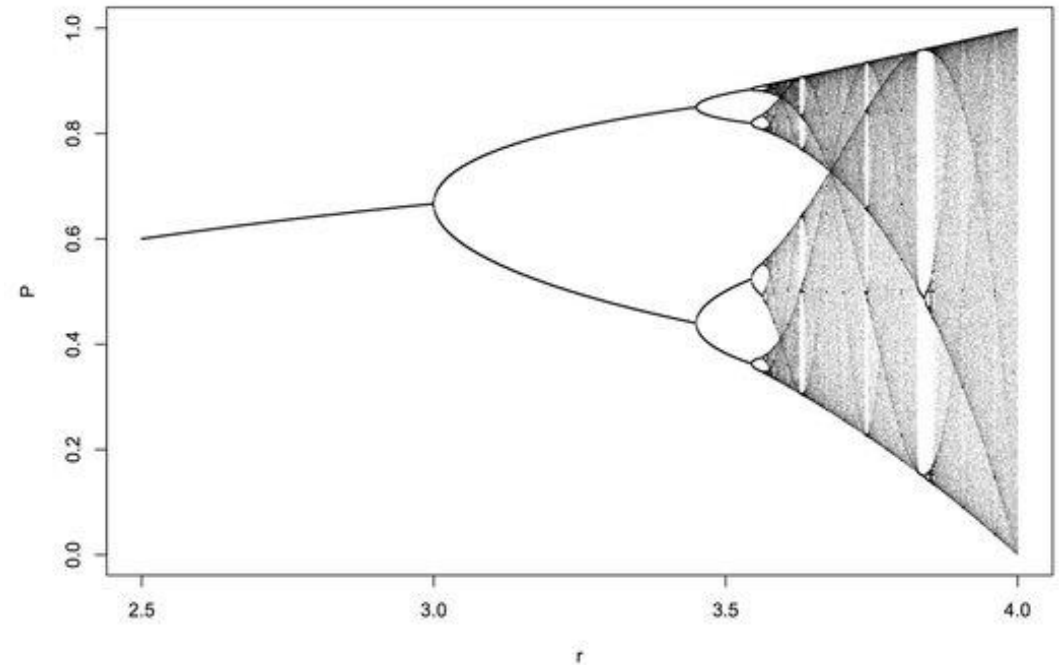
End if

End for

End

Rodzaje ruchów w JS

- ruch z prądem oceanu – ruch w kierunku wyznaczonym przez uśrednienie wektorów od każdej meduzy do meduzy o aktualnie najlepszym położeniu
- ruch pasywny (typ A w artykule) – losowa perturbacja wokół aktualnego położenia
- ruch aktywny (typ B w artykule) – ruch na prostej do losowo wybranej innej meduzy, w kierunku zgodnym, gdy jej położenie jest lepsze i w kierunku przeciwnym, gdy jej położenie jest gorsze



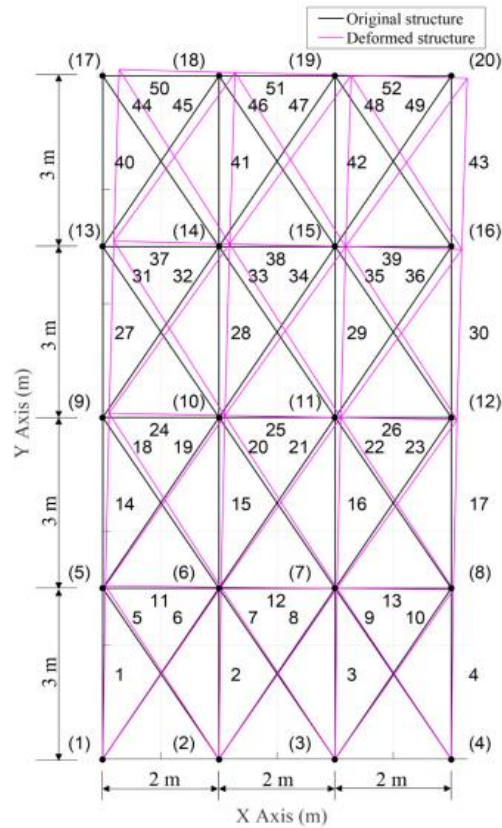
Tzw. Bifurcation Diagram dla mapy logistycznej

$$x_{n+1} = rx_n (1 - x_n)$$

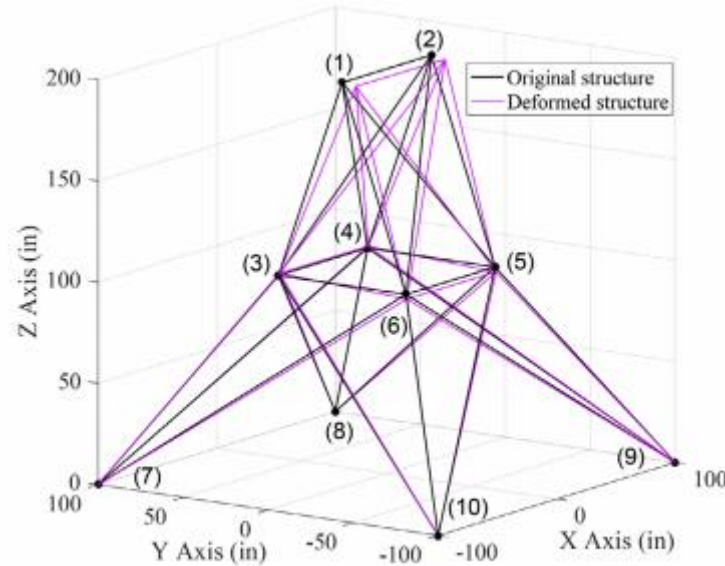
Subiektywa opinia o artykule JS

- +Rozbudowany przegląd literatury
- Niepotrzebne zastępowanie terminów (meduza, plankton)
- Za długi opis meduz i odwołania do nieistotnych artykułów biologicznych
- +Jasny opis inspiracji poszczególnych operacji matematycznych
- +Nowe mechanizmy aktualizacji pozycji (prąd morski, przełączanie między rodzajami ruchu, inicjalizacja mapą logistyczną)
- + - Estetyczne, ale nie zawsze informatywne obrazki
- +Jasny, matematyczny opis algorytmu
- +Dostępność kodu źródłowego (MATLAB)
- +Benchmarki dla wielu funkcji testowych (75, od 2 do 50 wymiarów) i licznych innych metaheurystyk (10)
- +Użycie testów statystycznych w benchmarkach
- Testowanie różnych wartości parametrów wewnętrznych (ukrytych przed użytkownikiem), strojenie pod benchmarki
- + -Testowanie wielu sposobów inicjalizacji, ale strojenie pod benchmarki
- +Dokładne podawanie parametrów innych algorytmów i specyfikacji użytego komputera
- +Dokładne podawanie wyników testów (tabele, średnie, odchylenia standardowe, czasy)
- Dziwne przeplatanie tekstu z tabelami w drugiej połowie artykułu
- +Ciekawe testy z optymalizacją strukturalną wież

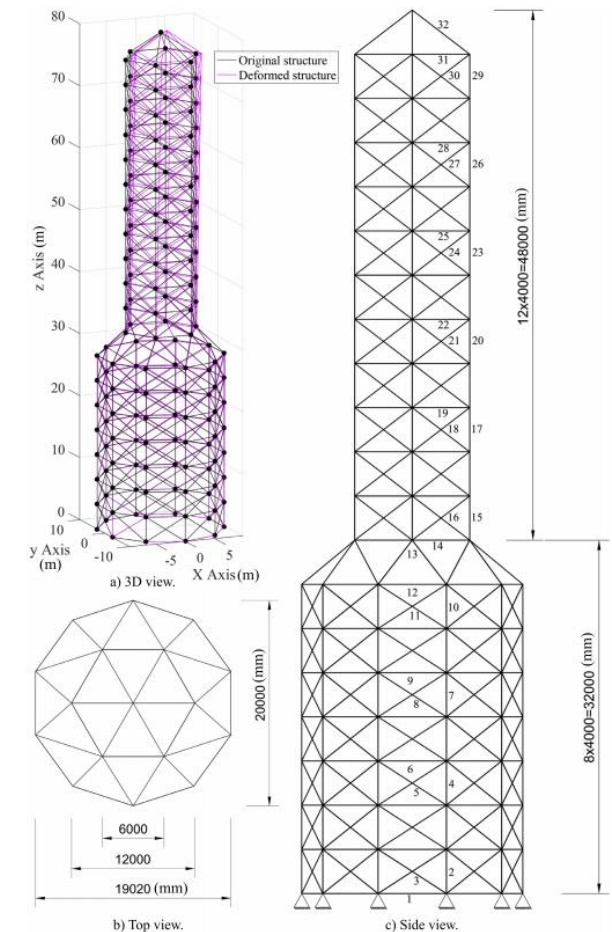
Optymalizacja strukturalna wież



a) Schematic of 52-bar tower.



a) Schematic of 25-bar tower.



Wartościowe metaheurystyki

- Tabu Search
- Ant Colony Optimization
- Genetic Algorithms
- Memetic Algorithms
- Approximation Algorithms
- Variable Neighborhood Search
- Greedy Randomized Adaptive Search
- Pilot Method
- Matheuristics



Linki do artykułów

- [Kirkpatrick, Gelatt Jr., Vecchi \(1983\)](#)
- [Laarhoven and Aarts \(1987\)](#)
- [Eberhart, Kennedy \(1995\)](#)
- [Zambrano-Bigiarini, Clerc, Rojas \(2011\)](#)
- [Pedersen \(2010\)](#)
- [Sörensen \(2013\)](#)
- [Geem, Kim, Loganathan \(2001\)](#)
- [Weyland \(2010\)](#)
- [Chou, Truong \(2020\)](#)

