

Algorytm świetlika
– charakterystyka, własności aplikacyjne
i usprawnienia

Szymon Łukasik

Instytut Badań Systemowych PAN

Katedra Automatyki i Techniki Informatycznej Politechniki Krakowskiej

Wprowadzenie

- Algorytm świetlika (ang. Firefly Algorithm, w skrócie FA) jest nową metaheurystyką rojową opracowaną w 2008 roku przez prof. Xin-She Yanga. Jest ona inspirowana zachowaniem społecznym świetlików (owadów z rodziny Lampyridae, w Polsce często określanych mianem robaczków świętojańskich) i fenomenem ich bioluminescencyjnej komunikacji.
- W zamierzeniu algorytm jest dedykowany do zagadnień optymalizacji ciągłej z ograniczeniami, gdzie rozważany jest problem minimalizacji funkcji kosztu $f(x)$, tj. znalezienie takiego x^* że:

$$f(x^*) = \min_{x \in S} f(x)$$

przy czym: $S \subset \mathbb{R}^n$.

- Niedawno pokazano również że algorytm świetlika może być z powodzeniem użyty także w problemach optymalizacji kombinatorycznej (przykład: przepływowe szeregowanie zadań).

FA – podstawy działania

- Załóżmy że istnieje rój m agentów (światełek) rozwiązujących iteracyjnie problem optymalizacji, przy czym x_i reprezentuje rozwiązanie dla światełka i w iteracji k , a $f(x_i)$ oznacza jego koszt.
- Każdy światełek posiada wyróżniającą go atrakcyjność β która określa jak silnie przyciąga on innych członków roju.
- Jako atrakcyjność światełka powinno się użyć malejącej funkcji odległości $r_j = d(x_i, x_j)$ do wybranego światełka j , np. jak sugeruje Yang funkcji:

$$\beta = \beta_0 e^{-\gamma r_j^2}$$

gdzie β_0 i γ są ustalonymi na wstępie parametrami algorytmu, odpowiednio: maksimum atrakcyjności i współczynnikiem absorpcji.

Ruch świetlika

- Każdy członek roju jest charakteryzowany przez świetlność I_i która może być bezpośrednio wyrażona jako odwrotność wartości funkcji kosztu $f(x_i)$.
- Początkowo wszystkie świetliki są rozmieszczone w S (losowo bądź z użyciem pewnej deterministycznej strategii).
- By efektywnie eksplorować przestrzeń rozwiązań przyjmuje się że każdy świetlik i zmienia swoje położenie iteracyjnie, biorąc pod uwagę dwa czynniki: atrakcyjność innych członków roju o większej świetlności $I_j > I_i$, $\forall j = 1, \dots, m, j \neq i$ która maleje wraz z odległością oraz krok losowy u_i .
- Dla najjaśniejszego świetlika stosuje się jedynie wyżej wymieniony krok losowy.

FA w pseudokodzie

Wejście: $f(z)$, $S \subset \mathbb{R}^n$ {funkcja kosztu}, $S = [a_k, b_k]$, $\forall k = 1, \dots, n$ {ograniczenia}
 m , β_0 , γ , $\min u_i$, $\max u_i$ {parametry algorytmu}

Wyjście: $x_{i^{min}}$

begin

repeat

$i^{min} = \operatorname{argmin}_i f(x_i)$, $x_{i^{min}} = \operatorname{argmin}_{x_i} f(x_i)$

for $i=1$ **to** m **and** $i \neq i^{min}$ **do**

for $j=1$ **to** m **do**

if $f(x_j) < f(x_i)$ **then**

$r_j = \text{Calculate_Distance}(x_i, x_j)$

$\beta = \beta_0 e^{-r_j^2}$

$u_i = \text{Generate_Random_Vector}(\min u_i, \max u_i)$

for $k=1$ **to** n **do** $x_{i,k} = (1 - \beta)x_{i,k} + \beta x_{j,k} + u_{i,k}$

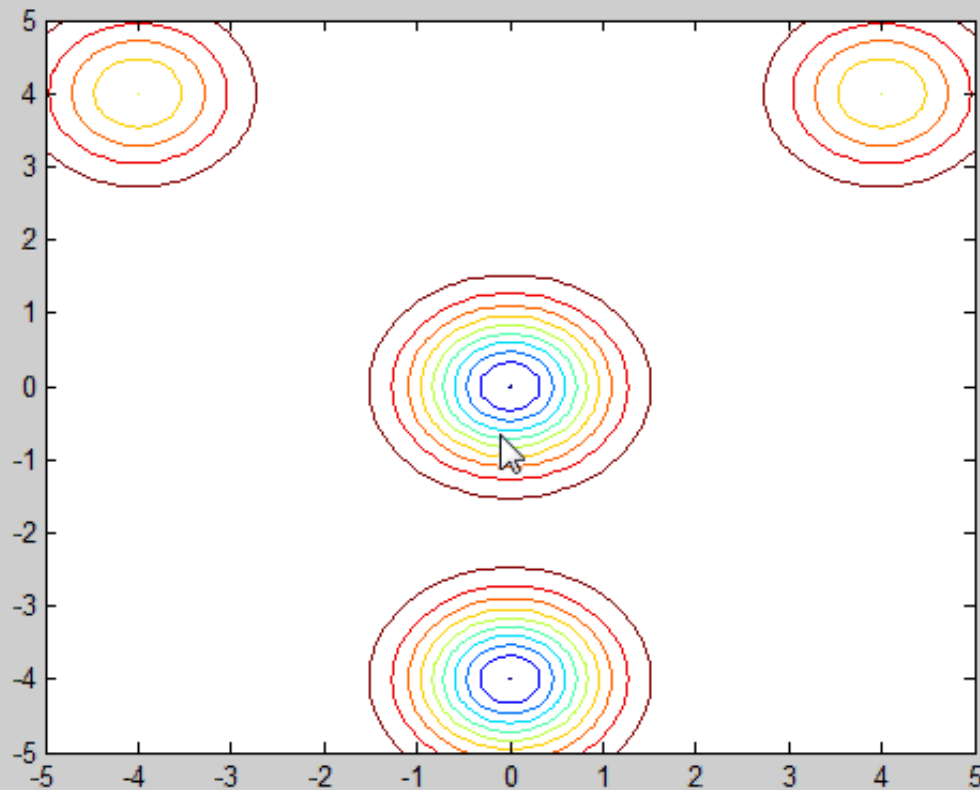
$u_{i^{min}} = \text{Generate_Random_Vector}(\min u_i, \max u_i)$

for $k=1$ **to** n **do** $x_{i^{min},k} = x_{i^{min},k} + u_{i^{min},k}$

until *stop condition false*

end

FA w akcji: funkcja 'four peaks'



Parametry algorytmu

- Można zauważyć że FA posiada trzy parametry określające wpływ innych rozwiązań i kroku losowego:
 - Maksimum atrakcyjności $\beta_0 \in [0, 1]$ determinuje atrakcyjność dla $r_j = 0$ (0 jest równoważne rozproszonemu losowemu przeszukiwaniu, 1 pełnej/maksymalnej zależności od innych członków roju).
 - Współczynnik absorpcji γ określa zmienność atrakcyjności wraz ze zwiększającą się odległością od świetlika z którym bieżący agent się komunikuje (0 oznacza brak zmienności tj. stałą atrakcyjność, $\rightarrow \infty$ to losowe przeszukiwanie).
 - Dolne i górne ograniczenie ($\min u_i, \max u_i$) określają zakres wartości kroku losowego.
- Należy także określić rozmiar roju m (warto tu odnotować, że FA posiada złożoność obliczeniową $O(m^2)$).

Uwagi techniczne

- Yang w klasycznym schemacie algorytmu proponuje użyć ustalonej wartości współczynnika absorpcji γ zależnej od „charakterystycznej długości rozważanego układu” (w przedziale: 0,01 do 100) i losowego kroku w postaci:

$$u_{i,k} = \alpha(rand - 1/2)$$

z dodatkowym skalowaniem wzdłuż wybranych wymiarów (gdy jest ono wymagane) oraz z $rand \sim U(0,1)$.

- Zbadaliśmy możliwość zastosowania zamiast ustalonej wartości współczynnika absorpcji jego adaptacyjnej postaci danej wzorem:

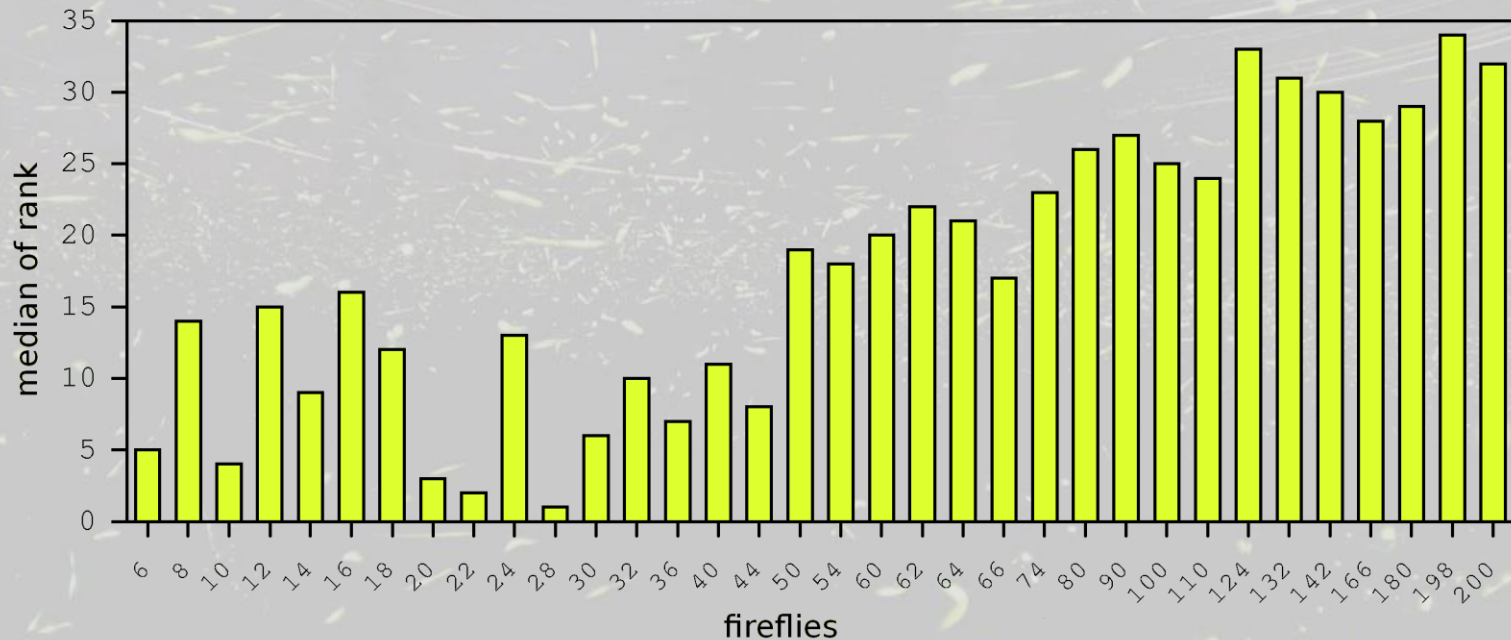
$$\gamma = \frac{\gamma_0}{r_{max}} \quad \text{lub} \quad \gamma = \frac{\gamma_0}{r_{max}^2}$$

z $\gamma_0 \in [0, 1]$ i $r_{max} = \max d(x_i, x_j), \forall x_i, x_j \in S$. Opcja ta wydaje się godna rozważenia (zwłaszcza dla $\gamma_0 \in [0.7 ; 1.0]$).

Wyniki eksperymentów

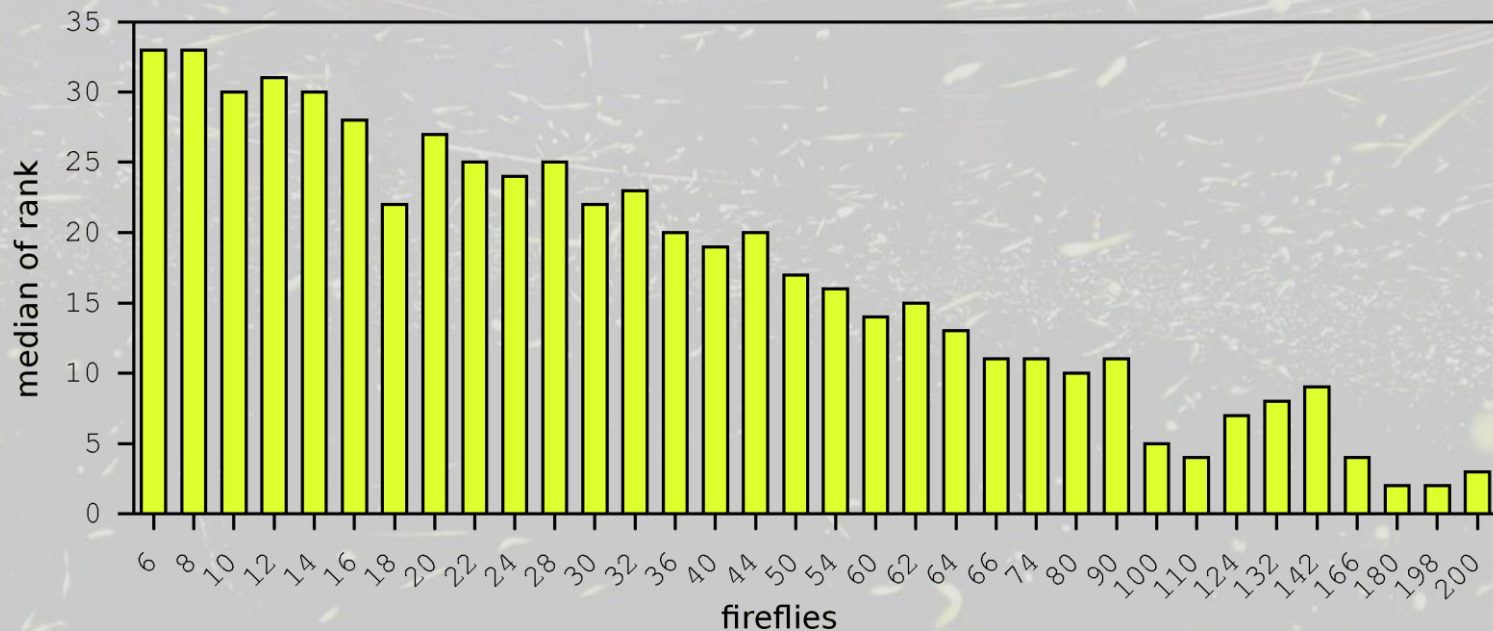
- Funkcjonowanie algorytmu zostało przebadane dla zestawu 14 dobrze znanych funkcji – problemów testowych z zakresu optymalizacji ciągłej (np. *himmelblau* 2, *schwefel* 10, *easom* 2, *rastrigin* 20, *griewank* 5, *rosenbrock* 4...)
- Wszystkie testy zostały przeprowadzone dla ustalonej liczby iteracji algorytmu i powtórzone w 100 niezależnych próbach.
- Jako że w/w problemy charakteryzują różne skale wartości funkcji kosztu w celu ewaluacji różnych wariantów FA użyto ich oceny rankingowej, zamiast bezpośredniej analizy wskaźnika $|f(x^*) - f(x_{i,min})|$. Oznacza to, że każdy problem rozważany było osobno, a warianty FA dla niego użyte były szeregowane zgodnie z jakością uzyskanego rozwiązania. Do ostatecznego porównania użyto median tak uzyskanych pozycji rankingowych.

Rozmiar populacji (grupa 1)



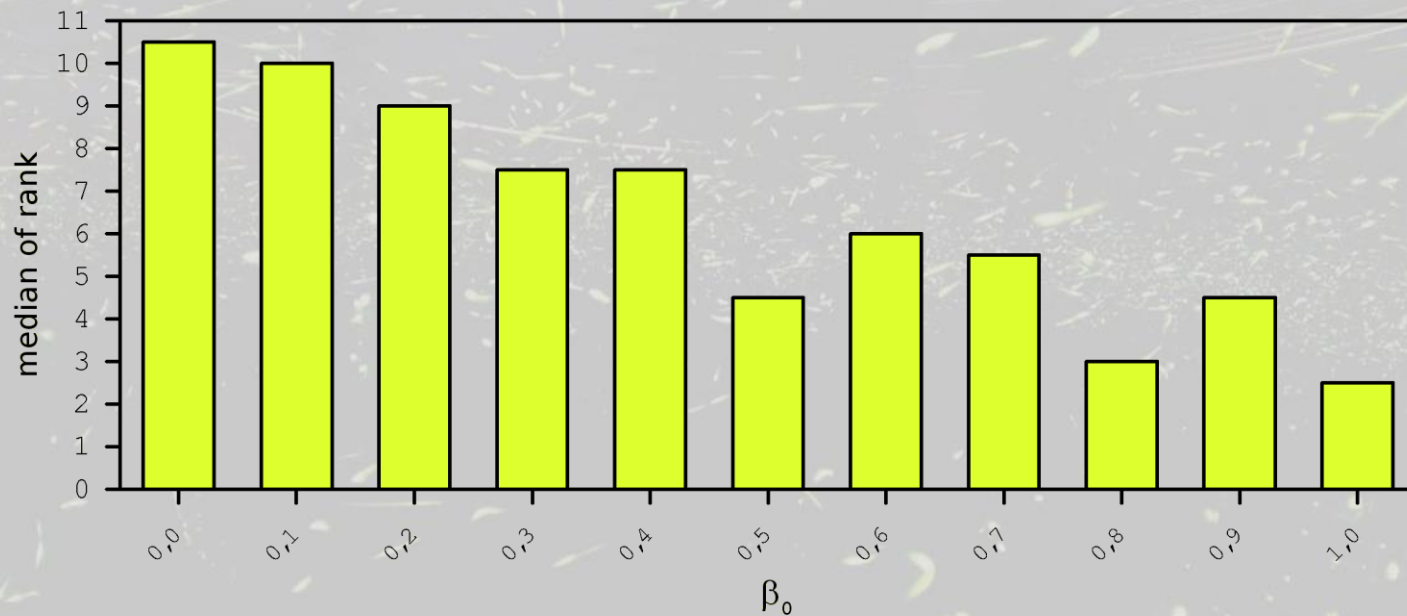
Mediana pozycji rankingowych dla zmiennego rozmiaru populacji
(funkcja *sphere 3*)

Rozmiar populacji (grupa 2)



Mediana pozycji rankingowych dla zmiennego rozmiaru populacji
(schwefel 10, easom 2, rastrigin 20, hartman 3&6, michalewicz 5,
powersum 4)

Maksimum atrakcyjności



Mediana pozycji rankingowych dla zmiennego maksimum funkcji atrakcyjności

Porównanie z algorytmem roju cząstek

- FA został porównany ze znajdującym się na znacznie wyższym stopniu rozwoju algorytmem PSO, zaimplementowanym ze współczynnikiem ograniczeń i zestawem rozsądnie zdefiniowanych parametrów (Schutte & Groenwold, 2005).
- Oba algorytmy zostały wykonane z tym samym rozmiarem populacji agentów $m = 40$, liczbą iteracji $l = 250$ a test powtórzono 100 razy, by uzyskane wyniki można było uznać za reprezentatywne.
- Użyto 9 wariantów FA ze zmiennym $\alpha \in \{0,001; 0,01; 0,1\}$ i $\gamma \in \{0,1; 1, 10\}$ zaś przedmiotem rozważań był zarówno najlepszy wariant (biorąc pod uwagę wszystkie problemy), jak i najlepszy wynik osiągnięty przez FA dla każdego problemu.
- Wyniki:
 - Najlepsza konfiguracja FA vs PSO: **4–11**
 - Najlepszy wynik uzyskany przez dowolny wariant FA vs PSO **6–9**

Przykładowe modyfikacje

- Yang proponuje w swojej najnowszej publikacji dotyczącej FA użycie zamiast prostego kroku losowego generowanego z rozkładu jednostajnego kroku opartego o generator rozkładu Lévy'ego :

$$u_{i,k} = \alpha \operatorname{sgn} \left[\operatorname{rand} - \frac{1}{2} \right] \times Lévy$$

- Sugeruje także zastosowanie $\lambda = 1.5$ jako parametru generatora rozkładu i określa nowy wariant FA mianem Lévy Firefly Algorithm (LFA).
- Yang używa także FA jako jeden ze składników nowo opracowanej metody heurystycznej zwanej strategią orła (ang. Eagle Strategy).

Dalsze badania

- Obiecującym pomysłem wydaje się być zrównoleglenie algorytmu – w sposób asynchroniczny, z przechowywaniem położeń i wartości funkcji kosztu w pamięci dzielonej.
- Hybrydyzacja FA z wybraną strategią lokalnego przeszukiwania – w szczególności z uogólnionym lub szybkim symulowanym wyżarzaniem lub CMA-ES – powinna przynieść znaczny wzrost jakości otrzymywanych wyników.
- Dalszych prac wymaga adaptacja wartości współczynnika absorpcji do rozważanego zagadnienia optymalizacji.
- W końcu algorytm ten powinien doczekać się rzetelnego porównania z PSO (zwłaszcza z wariantem FIPSO) opartego o *benchmarki* GECCO/CEC.
- Zastosowanie w optymalizacji kombinatorycznej – na podstawie doświadczeń z prac nad DPSO.



Dziękuję Państwu za uwagę!

Ważniejsze pozycje bibliograficzne

1. X.-S. Yang, "**Nature-Inspired Metaheuristic Algorithms**", Luniver Press, 2008.
2. X.-S. Yang, "**Engineering Optimization: An Introduction with Metaheuristic Applications**", Wiley, 2010.
3. M. K. Sayadi, R. Ramezani, N. Ghaffari-Nasab, "**A discrete firefly meta-heuristic with local search for makespan minimization in permutation flow shop scheduling problems**", International Journal of Industrial Engineering Computations, Vol. 1, 2010.
4. S. Łukasik, S. Żak, "**Firefly Algorithm for Continuous Constrained Optimization Tasks**", Lecture Notes in Artificial Intelligence, vol. 5796, 2009.
5. K.N. Krishnanand, D. Ghose. "**Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions**", Swarm Intelligence, vol. 3, 2009.
6. K.N. Krishnanand, D. Ghose, "**Glowworm swarm based optimization algorithm for multimodal functions with collective robotics applications**", Multi-agent and Grid Systems, vol. 2, 2006.