Firefly Algorithm for Continuous Constrained Optimization Tasks

Szymon Łukasik*   Sławomir Żak*

October 5, 2009

*Polish Academy of Sciences, Systems Research Institute

Firefly Algorithm (FA) is an optimization technique, developed recently by Xin-She Yang at Cambridge University[1]. It is inspired by social behavior of fireflies and the phenomenon of bioluminescent communication.

Our paper is devoted to the detailed description of the existing algorithm. We give some suggestions for extending the simple scheme of the algorithm, present some results of the performed experimental parameter studies and a comparison with existing Particle Swarm Optimization strategy based on existing benchmark instances.

The algorithm is considered in the **continuous constrained (bounded) optimization problem** setting where the task is to minimize cost function $f(x)$ for $x \in S \subset \mathbb{R}^n$ i.e. find x^* such as:

$$f(x^*) = \min_{x \in S} f(x) \ . \quad (1)$$

[1] See: "Nature-Inspired Metaheuristic Algorithms" by Xin-She Yang (Luniver Press, 2008)

FA concept

- Assume that there exists a **swarm of m agents (fireflies)** solving optimization problem iteratively and x_i represents a solution for a firefly i in algorithm's iteration k, whereas $f(x_i)$ denotes its cost.
- Each firefly has its **distinctive attractiveness** β which implies how strong it attracts other members of the swarm. As a firefly attractiveness one should select any monotonically decreasing function of the distance $r_j = d(x_i, x_j)$ to the chosen firefly j, e.g. as Yang suggests, the exponential function:

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

  where β_0 and γ are predetermined algorithm **parameters**: maximum attractiveness value and absorption coefficient, respectively.

FA concept (continued)

- Every member of the swarm is characterized by its **light intensity** I_i which can be directly expressed as a inverse of a cost function $f(x_i)$.
- Initially all fireflies are dislocated in S (randomly or employing some deterministic strategy).
- To effectively explore considered search space S it is assumed that each firefly i is changing its position iteratively taking into account two factors: **attractiveness** of other swarm members with higher light intensity i.e. $I_j > I_i, \forall j = 1, ...m, j \neq i$ which is varying across distance and a fixed **random step vector** u_i.
- If no brighter firefly can be found only the randomized step is being used.

FA in pseudocode

Input:

$f(z)$, $z = [z_1, z_2, ..., z_n]^T$ {cost function}, $S = [a_k, b_k], \forall k = 1, ..., n$ {constraints}

m, β_0, γ, $\min u_i$, $\max u_i$ {algorithm's parameters}

Output:

$x_{i^{min}}$

begin

  repeat

    $i^{min} \leftarrow \arg\min_i f(x_i)$, $x_{i^{min}} \leftarrow \arg\min_{x_i} f(x_i)$

    for *i=1* **to** *m* **do**

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

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

          $r_j \leftarrow$ `Calculate_Distance` (x_i, x_j)

          $\beta \leftarrow \beta_0 e^{-\gamma r_j}$

          $u_i \leftarrow$ `Generate_Random_Vector` $(\min u_i, \max u_i)$

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

    $u_{i^{min}} \leftarrow$ `Generate_Random_Vector` $(\min u_i, \max u_i)$

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

  until *stop condition true*

end

Example: Four Peaks

Technical Details

There are three parameters which control ratio of the influence of other solutions and the random step:

- **Maximum value** $\beta_0 \in [0, 1]$ determines the **attractiveness at** $r_j = 0$ (0 $\rightarrow$ distributed random search, 1 $\rightarrow$ total dependence).
- **Absorption coefficient** γ controls the **variation of attractiveness** with increasing distance from communicated firefly (0 $\rightarrow$ no variation or constant attractiveness, ∞ $\rightarrow$ complete random search).
- **Lower and upper bounds** ($\min u_i$, $\max u_i$) are put on the random step.

One have to choose as well suitable **population size** m (note that FA has **computational complexity** of $O(m^2)$.

To conclude: it would be desirable to possess some guidelines for algorithm's parameters and/or make them less problem-dependent.

Our proposals

- Instead of fixed random step size it is suggested here to define random vector as **a fraction of firefly distance to search space boundaries**:

$$u_{i,k} = \begin{cases} \alpha\, rand_2(b_k - x_{i,k}) & if \;\; \mathrm{sgn}(rand_1 - 0.5) < 0 \\ -\alpha\, rand_2(x_{i,k} - a_k) & if \;\; \mathrm{sgn}(rand_1 - 0.5) \geq 0 \end{cases} \tag{3}$$

with two uniform random numbers $rand_1, rand_2 \sim U(0,1)$ and $\alpha \in [0,1]$.

- **Customized absorption coefficient** could be based on the "characteristic length" of the optimized search space. It is proposed here to use:

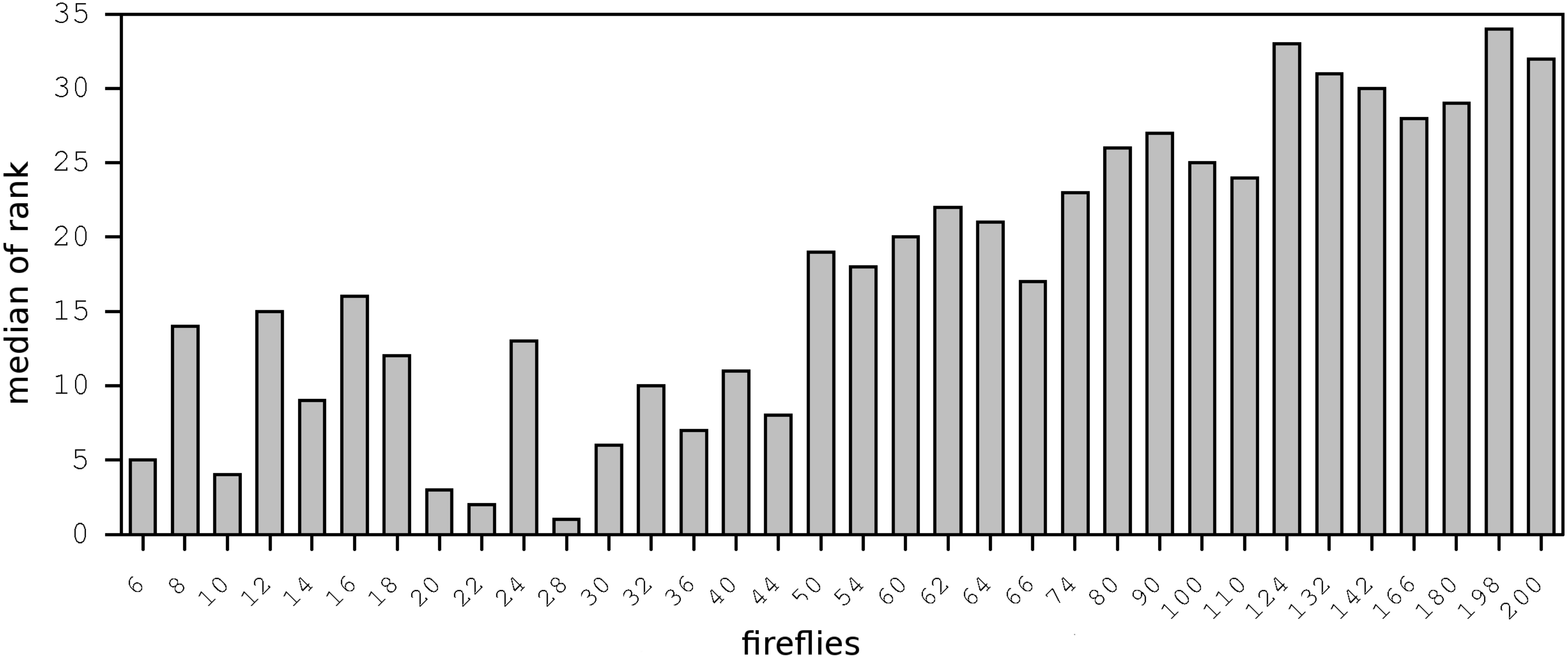
$$\gamma = \frac{\gamma_0}{r_{max}} \text{ or } \gamma = \frac{\gamma_0}{r_{max}^2} \tag{4}$$

wheras $\gamma_0 \in [0,1]$, $r_{max} = \max d(x_i, x_j), \forall x_i, x_j \in S$.

Experimental setup

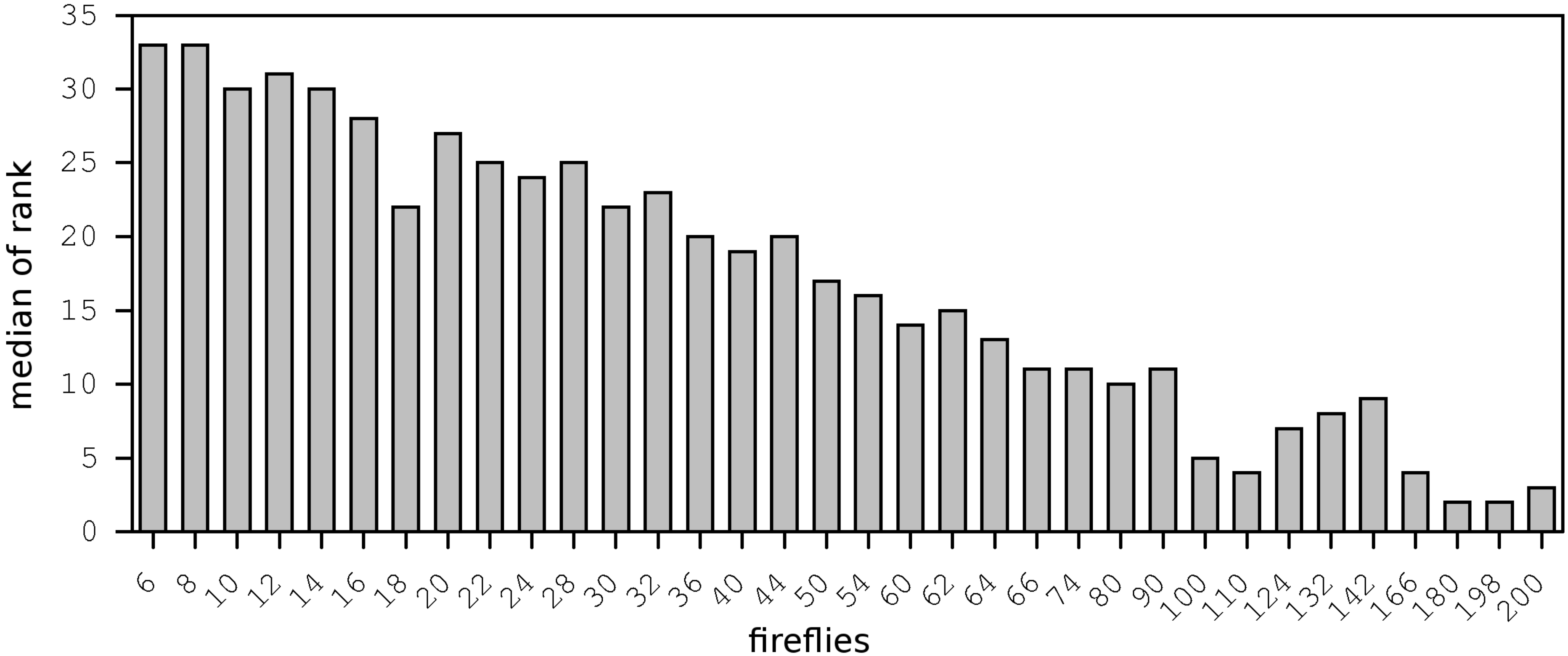
- Algorithm performance was tested for a set of **14** well known continuous optimization **benchmark problems**.
- All tests were conducted for a fixed number of algorithm iterations l and repeated in **100 independent trials**.
- As problems are characterized by different scales on the cost function it was more convenient to use **ranking** of different algorithm's variants instead of direct analysis of quality indexes $\left|f_{min} - f(x_{i^{min}})\right|$. It means that each problem was considered separately with tested configurations being ranked by their performance. Then the final comparison was carried out using **medians of obtained ranks**.

Parameter Studies 1: Population Size



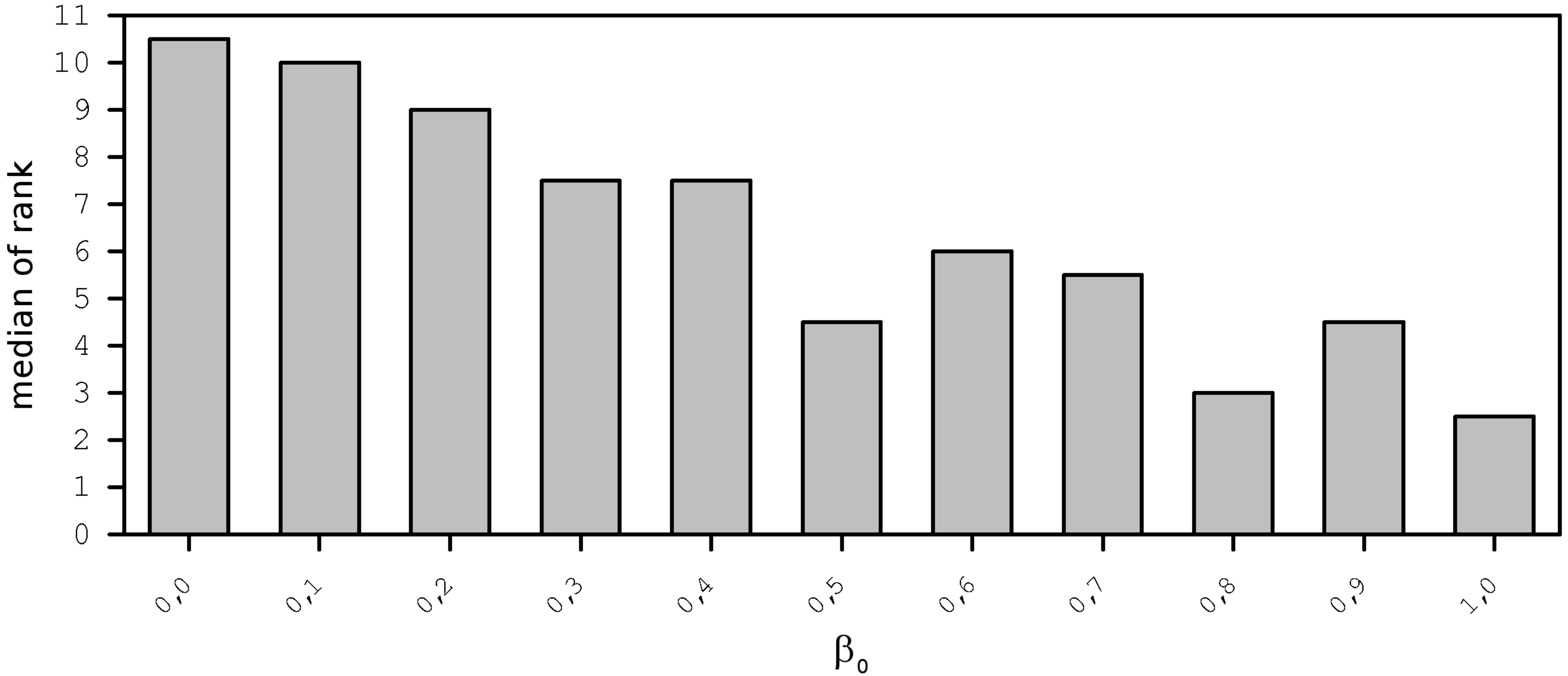
Median of performance ranks for varying population size
(Sphere function)

Parameter Studies 1: Population Size



Median of performance ranks for varying population size
(problems no 2, 3, 4, 8, 9, 13, 14)

PS 2: Maximum of Attractiveness Function



Median of performance ranks with varying maximum of attractiveness function

PS 3: Absorption Coefficient and Random Step

- Maximum attractiveness $\beta_0 = 1$ was used, with population size $m = 40$ and iteration number $l = 250$. Firefly Algorithm variants with $\alpha = \{0.001, 0.01, 0.1\}$ and $\gamma = \{0.1, 1.0, 10.0\}$ were tested. Additionally two problem-related techniques of obtaining absorption coefficient were considered (with $\gamma_0 = \{0.1, 0.2, ..., 1.0\}$), so the overall number of examined configurations reached **75**.

- Obtained results indicate that for the examined optimization problems variants of the algorithm with $\alpha = 0.01$ are the best in terms of performance. Furthermore it could be advisable to use **adaptable absorption coefficient** according with $\gamma_0 = 0.8$ and r_{max} as this configuration achieved best results in the course of executed test runs. Although proposed technique of γ adaptation in individual cases often performs worse than fixed γ values it has an advantage to be automatic and “tailored” to the considered problem.

Comparison with PSO

- Experiments involved a performance comparison of Firefly Algorithm with **Particle Swarm Optimization** algorithm defined with constriction factor and the parameters set suggested by Schutte and Groenwold in 2005[2].
- Both algorithms were executed with the same population size $m = 40$, iteration number $l = 250$ and the test was repeated 100 times for its results to be representative.
- Firefly Algorithm was found to be **outperformed repeatedly** by Particle Swarm Optimizer (PSO performed better for **11** benchmark instances out of 14 being used). It was also **less stable** in terms of standard deviation. It is important to observe though that the advantage of PSO is vanishing significantly (to **8** instances for which PSO performed better) when one relates it to the best configuration of firefly inspired heuristic algorithm.

[2] See: "A Study of Global Optimization Using Particle Swarms" by Jacob F. Schutte and Albert A. Groenwold (Journal of Global Optimization, vol. 31, 2005)

Conclusion

- Firefly Algorithm described here could be considered as an unconventional swarm-based heuristic algorithm for constrained optimization tasks and perceived as a kind of **“position-based PSO”**.
- At the current level of development the algorithm offers **worse performance** when compared with PSO.
- We tried to derive some **coherent suggestions** considering population size and maximum of absorption coefficient.
- The algorithm could benefit from additional research in the **adaptive establishment** of absorption coefficient and random step size.
- Some additional features like **decreasing random step size** and more sophisticated procedure of **initial solution generation** could bring further improvements in the algorithm performance.

Thank you for your attention!