**Lecture Notes in Management Science** (2012) Vol. **4**: 30–36
4th International Conference on Applied Operational Research, Proceedings

ISSN 2008-0050 (Print), ISSN 1927-0097 (Online)

# A computationally efficient modelling-to-generate-alternatives method using the firefly algorithm

Raha Imanirad and Julian Scott Yeomans

*OMIS Area, Schulich School of Business, York University, Toronto, Canada*
*syeomans@schulich.yorku.ca*

**Abstract.** In solving many practical mathematical programming applications, it is preferable to formulate numerous quantifiably good alternatives that provide very different perspectives to the problem. This is because decision-making typically involves complex problems that are riddled with incompatible and inconsistent performance objectives and possess competing design requirements which are very difficult – if not impossible – to quantify and capture at the time that the supporting decision models are constructed. There are invariably unmodelled design issues, not apparent at the time of model construction, which can greatly impact the acceptability of the model's solutions. Consequently, it is preferable to generate several alternatives that provide multiple, disparate perspectives to the problem. These alternatives should possess near-optimal objective measures with respect to all known modelled objective(s), but be fundamentally different from each other in terms of the system structures characterized by their decision variables. This solution approach is referred to as modelling-to-generate-alternatives (MGA). This study demonstrates how the biologically-inspired, Firefly algorithm can be used to efficiently create multiple solution alternatives that both satisfy required system performance criteria and yet are maximally different in their decision spaces.

**Keywords:** firefly algorithm; modelling-to-generate-alternatives; nature-inspired algorithms

## Introduction

Typical "real world" decision-making applications involve complex problems that possess requirements which are very difficult to incorporate into supporting decision models and tend to be riddled with competing performance objectives. While mathematically optimal solutions can provide the best solutions to the modelled problems, they are generally not the best answers to the underlying real problems as there are invariably unquantified issues and unmodelled objectives not apparent at the time of model construction. Consequently, it is preferable to generate a number of different alternatives that provide multiple, disparate perspectives to the particular problem (Yeomans & Gunalay 2011). Preferably these alternatives should all possess good (i.e. near-optimal) objective measures with respect to the modelled objective(s), but be as fundamentally different as possible from each other in terms of the system structures characterized by their decision variables. In response to this option creation requirement, several approaches collectively referred to as *modelling-to-generate-alternatives* (MGA) have been developed (Loughlin *et al*. 2001). The primary motivation behind MGA is to produce a manageably small set of alternatives that are good with respect to modelled objectives yet as different as possible from each other in the decision space. In so doing, the resulting alternative solution set is likely to provide truly different choices that all perform somewhat similarly with respect to the modelled objectives, yet very differently with respect to the unmodelled issues.

In this paper, it is shown how to efficiently generate a set of maximally different solution alternatives by implementing a modified version of the Firefly Algorithm (FA) of Yang (2010). For optimization purposes, Yang (2010) has demonstrated that FA is more computationally efficient than such commonly-used metaheuristics as genetic algorithms, simulated annealing, and enhanced particle swarm optimization. Hence, this new innovative MGA approach using FA is a very computationally efficient procedure. This study illustrates the efficacy of the MGA capabilities of this new FA approach procedure in constructing multiple, good-but-very-different solution alternatives to a constrained optimization test problem.

## Firefly algorithm for function optimization

While this section provides a brief synopsis of the steps involved in the FA process, more specific details can be found in Yang (2010). The FA is a biologically-inspired metaheuristic that employs the following three idealized rules: (i) All fireflies are unisex so that one firefly will be attracted to other fireflies irrespective of their sex; (ii) Attractiveness between fireflies is proportional to their brightness, implying that for any two flashing fireflies, the less bright one will move towards the brighter one. Attractiveness and brightness both decrease as the distance between fireflies increases. If there is no brighter firefly within its visible vicinity, then a particular firefly will move randomly; and (iii) The brightness of a firefly is determined by the landscape of the objective function. Namely, for a maximization problem, the brightness can simply be proportional to the value of the objective function. Based upon these three rules, the basic steps of the FA are summarized within the pseudo-code of Figure 1.

```
Objective Function f(X), X = (x1, x2,…xd)
Generate the initial population of n fireflies, Xi, i = 1,
2,…,n
Light intensity Ii at Xi is determined by f(Xi)
Define the light absorption coefficient γ
while (t < MaxGeneration)
          for i = 1: n , all n fireflies
            for j = 1: n ,all n fireflies (inner loop)
              if (Ii < Ij), Move firefly i towards j; end if
              Vary attractiveness with distance r via e^(- γr)
            end for j
          end for i
          Rank the fireflies and find the current global
best solution g*
end while
Postprocess the results
```

**Fig. 1.** Pseudo Code of the Firefly Algorithm

In the FA, there are two important issues to resolve: the variation of light intensity and the formulation of attractiveness. For simplicity, it can always be assumed that the attractiveness of a firefly is determined by its brightness which in turn is associated with the encoded objective function. In the simplest case, the brightness of a firefly at a particular location $X$ would be its calculated objective value $F(X)$. However, the attractiveness, $\beta$, between fireflies is relative and will vary with the distance $r_{ij}$ between firefly $i$ and firefly $j$. In addition, light intensity decreases with the distance from its source, and light is also absorbed in the media, so the attractiveness should be allowed to vary with the degree of absorption. Consequently, the overall attractiveness of a firefly can be defined as $\beta = \beta_0 \exp(-\gamma r^2)$, where $\beta_0$ is the attractiveness at distance $r = 0$ and $\gamma$ is the fixed light absorption coefficient for a specific medium. If the distance $r_{ij}$ between any two fireflies $i$ and $j$ located at $X_i$ and $X_j$, respectively, is calculated using the Euclidean norm, then the movement of a firefly $i$ that is attracted to another more attractive (i.e. brighter) firefly $j$ is determined by $X_i = X_i + \beta_0 \exp(-\gamma(r_{ij})^2)(X_i - X_j) + \alpha\varepsilon_i$ .

In this expression of movement, the second term is due to the relative attraction and the third term is a randomization component. Yang (2010) indicates that $\alpha$ is a randomization parameter normally selected within the range [0,1] and $\varepsilon_i$ is a vector of random numbers drawn from either a Gaussian or uniform (generally [-0.5,0.5]) distribution. It should be pointed out that this expression is a random walk biased toward brighter fireflies and if $\beta_0 = 0$, it becomes a simple random walk. The parameter $\gamma$ characterizes the variation of the attractiveness and its value determines the speed of the algorithm's convergence. For most applications, $\gamma$ is typically set between 0.1 to 10 (Yang 2010). In any given optimization problem, for a very large number of fireflies $n >> k$ where $k$ is the number of local optima, the initial locations of the *n* fireflies should be distributed relatively uniformly throughout the entire search space. As the FA proceeds, the fireflies would converge into all of these local optima (including the global ones). By comparing the best solutions among all these optima, the global optima can easily be determined. Yang (2010) demonstrates that the FA will approach the global optima when $n \rightarrow \infty$ and the number of iterations $t$, is set so that $t >> 1$. In reality, the FA has been found to converge extremely quickly.

Two important limiting or asymptotic cases occur when $\gamma \rightarrow 0$ and when $\gamma \rightarrow \infty$. For $\gamma \rightarrow 0$, the attractiveness is constant $\beta = \beta_0$, which is equivalent to having a light intensity that does not decrease. Thus, a firefly would be visible anywhere within the solution domain. Hence, a single (usually global) optima can easily be reached. If the inner loop for $j$ in Figure 1 is removed and $X_j$ is replaced by the current global best $g^*$, then this implies that the FA becomes a special case of the accelerated particle swarm optimization (PSO) algorithm. Subsequently, the computational efficiency of this special case of the FA is equivalent to that of enhanced PSO. Conversely, when $\gamma \rightarrow \infty$, the attractiveness is essentially zero in the sightline of other fireflies. This is equivalent to the case where the fireflies randomly roam throughout a very thick foggy region. No other fireflies are visible and each firefly roams in a completely random fashion. This case corresponds to a completely random search method. As the FA operates between these two extremes, it is possible to adjust the parameters $\alpha$ and $\gamma$ so that the FA can outperform both the random search and the enhanced PSO algorithms. Furthermore, the FA can find both the global optima as well as the local optima concurrently which holds huge computational and efficiency advantages for MGA purposes (Yeomans and Gunalay 2011). Another additional advantage of the FA for MGA implementation is that different fireflies essentially work independently of each other and FA are thus better than genetic algorithms and PSO for MGA because the fireflies can aggregate more closely around each local optimum.

## Modelling to generate alternatives with the firefly algorithm

In most "real world" decision problems, there are numerous system objectives and requirements that are never made explicitly apparent or included at the decision formulation stage. Furthermore, it may never be possible to explicitly express all of the subjective considerations because there are frequently numerous incompatible, competing, adversarial stakeholder groups. Therefore most subjective aspects remain unquantified and unmodelled in the construction of any corresponding decision models. This is a common occurrence in situations where the final decisions are constructed based not only upon clearly stated and modelled objectives, but also upon fundamentally subjective, political and socio-economic goals and stakeholder preferences (Yeomans and Gunalay 2011). When unmodelled objectives and unquantified issues exist, different approaches are required in order to not only search the decision space for the noninferior set of solutions, but also to explore the decision space for inferior alternative solutions to the modelled problem.

In order to properly motivate an MGA search procedure, it is necessary to provide a more formal definition of the goals of the MGA process (Loughlin *et al*. 2001; Yeomans and Gunalay 2011). Suppose the optimal solution to an original mathematical model is $X^*$ with objective value $Z^* = F(X^*)$. The following model can then be solved to generate an alternative solution that is maximally different from $X^*$:

$$Max\ \Delta = \sum_i |X_i - X_i^*|$$

Subject to:

$$X \in D$$

$$|F(X) - Z^*| \leq T$$

where $\Delta$ is a difference function and $T$ is a target specified in relation to the original optimal function value $Z^*$. $T$ is a user-supplied value that represents how much of the inferior region is to be explored for alternative solutions. The FA-based MGA procedure is designed to generate a small number of good but maximally different alternatives by adjusting the value of $T$ and using the FA to solve the corresponding, new maximal difference problem instance. In this approach, subpopulations within the algorithm's overall population are established as the Fireflies collectively evolve toward different local optima in the solution space. Each desired solution alternative undergoes the common search procedure of the FA. The survival of solutions depends upon how well the solutions perform with respect to the modelled objective(s) and by how far away they are from all of the other previously generated alternatives in the decision space.

## Computational testing of the firefly algorithm used for MGA

The application of the MGA procedure will be illustrated using a spring design problem taken from Cagnina *et al*. (2008). The design of a tension and compression spring has frequently been employed as a benchmark problem for constrained engineering optimization problems (Cagnina *et al*. 2008). The problem involves three design variables: (i) $x_1$, the wire diameter, (ii) $x_2$, the coil diameter, and (iii) $x_3$, the length of the coil. The aim is to essentially minimize the weight subject to constraints on deflection, stress, surge frequency and geometry. The mathematical formulation for this test problem can be summarized as:

$$Min\ F(X) = x_1^2 x_2 (2 + x_3)$$

Subject to:

$$g_1(X) = 1 - \frac{x_2^3 x_3}{71785 x_1^4} \leq 0$$

$$g_2(X) = \frac{4x_2^2 - x_1x_2}{12566\left(x_1^3x_2 - x_1^4\right)} + \frac{1}{5108x_1^2} - 1 \le 0$$

$$g_3(X) = 1 - \frac{140.45x_1}{x_2^2x_3} \le 0$$

$$g_4(X) = \frac{x_1 + x_2}{1.5} - 1 \le 0$$

$$0.05 \le x_1 \le 2.0$$

$$0.25 \le x_2 \le 1.3$$

$$2.0 \le x_3 \le 15.0$$

For the design parameters employed in this specific problem formulation, Cagnina *et al*. (2008) provide a best solution of F($X$*) = 0.0127 with $X$* = (0.051690, 0.356750, 11.287126). However, as outlined earlier, planners generally prefer to be able to select from a set of "near-optimal" alternatives that significantly differ from each other in terms of the system structures characterized by their decision variables. In order to create this set of alternative planning options, extra target constraints that varied the value of $T$ were placed into this original formulation in order to force the generation of solutions that were maximally different from the initial optimal solution. The MGA difference model described in the previous section was used to produce the optimal solution and the 10 maximally different solutions shown in Table 1.

As described earlier, many problems are typically riddled with incongruent performance requirements that contain significant stochastic uncertainty that are also very difficult to quantify. Consequently, it is preferable to create several quantifiably good alternatives that concurrently provide very different perspectives to the potentially unmodelled performance design issues during the policy formulation stage. The unique performance features captured within these dissimilar alternatives can result in very different system performance with respect to the unmodelled issues, thereby incorporating the unmodelled issues into the actual solution process. This example has demonstrated how an MGA modelling perspective can be used to generate multiple, good policy alternatives via the computationally efficiently FA that satisfy required system performance criteria according to prespecified bounds and yet remain as maximally different from each other as possible in the decision space. In addition to its alternative generating capabilities, the MGA procedure has simultaneously performed exceedingly well with respect to its role in function optimization. It should be explicitly noted that the cost of the overall best solution produced by the MGA procedure is indistinguishable from the one determined in the much more straightforward function optimization approach of Cagnina *et al*. (2008).

**Table 1.** Objective Values and Solutions for the 11 Maximally Different Alternatives.

| *Increment* | *1% Increment Between Alternatives* | | | | *2.5% Increment Between Alternatives* | | | |
|---|---|---|---|---|---|---|---|---|
| | *F(X)* | $x_1$ | $x_2$ | $x_3$ | *F(X)* | $x_1$ | $x_2$ | $x_3$ |
| *Optimal* | 0.0127 | 0.05 | 0.3174 | 14.0324 | 0.0127 | 0.05 | 0.3174 | 14.0322 |
| *Alternative 1* | 0.0128 | 0.05 | 0.3164 | 14.1754 | 0.0128 | 0.05 | 0.3165 | 14.1598 |
| *Alternative 2* | 0.0128 | 0.0514 | 0.3472 | 12.0089 | 0.0131 | 0.05 | 0.3129 | 14.777 |
| *Alternative 3* | 0.0129 | 0.0529 | 0.3862 | 9.9684 | 0.0132 | 0.05 | 0.3167 | 14.6402 |
| *Alternative 4* | 0.013 | 0.0521 | 0.3656 | 11.0667 | 0.0140 | 0.0557 | 0.4307 | 9.5783 |
| *Alternative 5* | 0.0131 | 0.0527 | 0.3766 | 10.5179 | 0.0143 | 0.0542 | 0.4014 | 11.6481 |
| *Alternative 6* | 0.0134 | 0.05 | 0.3157 | 14.978 | 0.0146 | 0.0546 | 0.4247 | 10.7556 |
| *Alternative 7* | 0.0135 | 0.0524 | 0.3597 | 11.6966 | 0.0149 | 0.0562 | 0.438 | 11.1197 |
| *Alternative 8* | 0.0137 | 0.052 | 0.3629 | 12.1615 | 0.0152 | 0.0605 | 0.4836 | 8.9963 |
| *Alternative 9* | 0.0138 | 0.0523 | 0.348 | 13.3247 | 0.0156 | 0.0574 | 0.3841 | 14.5182 |
| *Alternative 10* | 0.0140 | 0.0535 | 0.3857 | 14.162 | 0.0159 | 0.0553 | 0.4072 | 15.0000 |

## Conclusions

In this paper, a procedure was presented that demonstrated how the computationally efficient FA could be used to generate multiple, maximally different, near-best alternatives. In this MGA capacity, the FA produces numerous solutions possessing the requisite problem characteristics, with each generated alternative providing a very different perspective. Since FA techniques can be adapted to solve a wide variety of problem types, the practicality of this MGA approach can clearly be extended into numerous disparate planning applications. These extensions will be studied in future research.

## References

Cagnina LC, Esquivel CA, and Coello CA (2008). Solving Engineering Optimization Problems with the Simple Constrained Particel Swarm Optimizer. Informatica 32: 319-326.

Loughlin DH, Ranjithan S.R, Brill ED and Baugh JW (2001). Genetic Algorithm Approaches for Addressing Unmodeled Objectives in Optimization Problems. Engineering Optimization 33(5): 549-569.

Yang XS (2010). Nature-Inspired Metaheuristic Algorithms 2nd Ed. Luniver Press. Frome, UK.

Yeomans JS and Gunalay Y (2011). Simulation-Optimization Techniques for Modelling to Generate Alternatives in Waste Management Planning. Journal of Applied Operational Research 3(1): 23-35.