A Chaotic Firefly Algorithm Applied to Reliability-Redundancy Optimization

Leandro dos Santos Coelho
Industrial and Systems Engineering Graduate Program
PPGEPS, Pontifical Catholic University of Paraná
Imaculada Conceição, 1155, Zip code 80215-901
Curitiba, Paraná, Brazil
leandro.coelho@pucpr.br

Diego Luís de Andrade Bernert
Industrial and Systems Engineering Graduate Program
PPGEPS, Pontifical Catholic University of Paraná
Imaculada Conceição, 1155, Zip code 80215-901
Curitiba, Paraná, Brazil
dbernert@gmail.com

Viviana Cocco Mariani
Department of Mechanical Engineering
PPGEM, Pontifical Catholic University of Paraná
Imaculada Conceição, 1155, Zip code 80215-901
Curitiba, Paraná, Brazil
viviana.mariani@pucpr.br

Abstract— **The reliability-redundancy allocation problem can be approached as a mixed-integer programming problem. It has been solved by using optimization techniques such as dynamic programming, integer programming, and mixed-integer non-linear programming. On the other hand, a broad class of meta-heuristics has been developed for reliability-redundancy optimization. Recently, a new meta-heuristics called firefly algorithm (FA) algorithm has emerged. The FA is a stochastic metaheuristic approach based on the idealized behavior of the flashing characteristics of fireflies. In FA, the flashing light can be formulated in such a way that it is associated with the objective function to be optimized, which makes it possible to formulate the firefly algorithm. This paper introduces a modified FA approach combined with chaotic sequences (FAC) applied to reliability-redundancy optimization. In this context, an example of mixed integer programming in reliability-redundancy design of an overspeed protection system for a gas turbine is evaluated. In this application domain, FAC was found to outperform the previously best-known solutions available.**

Keywords— **evolutionary algorithms, firefly optimization, chaotic sequences, optimization, reliability-redundancy optimization.**

I. Introduction

Redundancy optimization is a classical problem that has attracted considerable attention from the research community. A reliability-redundancy optimization problem can be formulated to use components, and levels-of-redundancy to maximize some objective function, given system-level constraints on reliability, cost, and/or weight. Reliability-redundancy optimization has been the subject of many studies [1],[2]. During the past two decades, numerous reliability design approaches based on optimization techniques [1]-[4] have been proposed.

Recently, many research activities have been devoted to design, implementation and validation of new efficient metaheuristic algorithms [5],[6]. Firefly algorithm (FA) is a new population-based metaheuristic approach developed by Xin-She Yang [7],[8], which is nature-inspired by behavior of the flashing characteristics of fireflies. In this context, the flashing light can be formulated in such a way that it is associated with the objective function to be optimized, which makes it possible to formulate the firefly algorithm.

On the other hand, in recent years, growing interests in chaos theory and its features have stimulated the studies of chaos applied in optimization algorithms design [9]-[12]. This paper proposes a combination of FA with chaotic sequences (FAC). To illustrate the power of the proposed FAC, a benchmark of reliability-redundancy optimization has been considered. The example is an overspeed protection system for a gas turbine [13]-[15]. Simulation results of the FA and the proposed FAC approach are compared with other optimization techniques presented in literature for a benchmark of reliability-redundancy optimization.

The remainder of this paper is organized as follows. In Section II, the reliability-redundancy optimization problem is introduced, while the characteristics of FA and FAC approaches are detailed in Section III. The problem formulation for reliability-redundancy optimization and its assumptions are given in Section IV. Moreover, Section IV also presents the simulation results for a benchmark problem of reliability-redundancy optimization. Finally, the conclusion and further research are discussed in Section V.

II. Reliability-Redundancy Optimization

In this work, the reliability-redundancy allocation problem of maximizing the system reliability subject to constraints can be formulated as

978-1-4244-7835-4/11/$26.00 ©2011 IEEE

$$\text{Maximize} \quad R_s = f(\boldsymbol{r}, \boldsymbol{n}), \tag{1}$$

$$\text{subject to} \quad g(\boldsymbol{r}, \boldsymbol{n}) \le l \tag{2}$$

$$0 \le r_i \le 1, \quad r_i \in \Re, \; n_i \in Z^+, \; 1 \le i \le m.$$

where R_s is the reliability of system, g is the set of constraint functions usually associated with system weight, volume and cost, $\boldsymbol{r} = (r_1, r_2, r_3, \ldots, r_m)$ is the vector of the component reliabilities for the system, $\boldsymbol{n} = (n_1, n_2, n_3, \ldots, n_m)$ is the vector of the redundancy allocation for the system; r_i and n_i are the reliability and the number of components in the ith subsystem respectively; $f(\cdot)$ is the objective function for the overall system reliability; and l is the resource limitation; m is the number of subsystems in the system. Our goal is to determine the number of components, and the components' reliability in each system so as to maximize the overall system reliability. The problem belongs to the category of constrained nonlinear mixed-integer optimization problems. The benchmark of reliability-redundancy optimization evaluated in this paper is formulated, which are outlined below.

A. Overspeed protection system for a gas turbine

The benchmark considered is an overspeed protection system for a gas turbine [13]-[15]. Overspeed detection is continuously provided by the electrical and mechanical systems. When an overspeed occurs, it is necessary to cut off the fuel supply using control valves [13].

This problem is formulated as the following mixed-integer nonlinear programming problem, i.e., the problem can be stated as

$$\text{Maximize} \quad f(\boldsymbol{r}, \boldsymbol{n}) = \prod_{i=1}^{m} \left[1 - (1 - r_i)^{n_i}\right], \tag{3}$$

subject to

$$g_1(\boldsymbol{r}, \boldsymbol{n}) = \sum_{i=1}^{m} v_i \cdot n_i^2 \le V \tag{4}$$

$$g_2(\boldsymbol{r}, \boldsymbol{n}) = \sum_{i=1}^{m} C(r_i) \cdot \left[n_i + e^{0.25 \cdot n_i}\right] \le C \tag{5}$$

$$g_3(\boldsymbol{r}, \boldsymbol{n}) = \sum_{i=1}^{m} w_i \cdot n_i \cdot e^{0.25 \cdot n_i} \le W \tag{6}$$

where $1 \le n_i \le 10$, $n_i \in Z^+$, where Z^+ is the space discrete of positive integers, $0.5 \le r_i \le 1\text{-}10^{-6}$, $r_i \in \Re$; v_i is the volume of each component in subsystem i; V is the upper limit on the sum of the subsystems' products of volume and weight; C is the upper limit on the system cost; $C(r_i) = a_i \cdot [-T/\ln(r_i)]^{b_i}$ is the cost of each component with reliability r_i at subsystem i; T is the operating time during which the component must not fail; and W is the upper limit on the weight of the system. The input parameters of the overspeed protection system for a gas turbine are shown in Table I.

TABLE I
DATA OF OVERSPEED PROTECTION SYSTEM

Stage	$10^5 \cdot a_i$	b_i	v_i	w_i	V	C	W	T
1	1.0	1.5	1	6	250	400	500	1000 h
2	2.3	1.5	2	6				
3	0.3	1.5	3	8				
4	2.3	1.5	2	7				

III. OPTIMIZATION ALGORITHM

This section describes the proposed FA. First, a brief overview of the FA is provided, and finally the modification procedures of the proposed FAC are presented.

A. Firefly algorithm

The FA is a metaheuristic algorithm, inspired by the flashing behavior of fireflies. Recent studies show that the FA is efficient in numerical optimization [16]-[18] and combinatorial optimization [19], and could outperform other metaheuristics [16].

Yang [7],[8] formulated this firefly algorithm by assuming: i) all fireflies are unisex, so that one firefly will be attracted to all other fireflies; ii) attractiveness is proportional to their brightness, and for any two fireflies, the less brighter one will attract to the brighter one; however, the brightness can decrease as their distance increases; and iii) if there are no fireflies brighter than a given firefly, it will move randomly. The brightness should be associated with the objective function.

As a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, we can now define the attractiveness β of a firefly in terms of Cartesian distance between firefly i and firefly j. In this case, the movement of a firefly i is attracted to another more attractive (brighter) firefly j is determined by [7]

$$x_i = x_i + \beta_0 e^{-\gamma \cdot r_{ij}^2}(x_j - x_i) + \alpha\left(rand - \frac{1}{2}\right) \tag{7}$$

where the second term is due to the attraction, where γ is the absorption coefficient, and β_0 is the attractiveness at $r = 0$. The third term is randomization with α being the randomization parameter. The value of *rand* is a random number generator uniformly distributed in [0, 1]. Based on comments in [7] for the implementation of classical FA was adopted $\beta_0 = 1$ and α generated with uniform distribution in range [0,1] in this paper. The procedure for implementing the FA can be summarized as the pseudo code (adapted of [7]) shown in Fig. 1.

```
Objective function of optimization problem f(x), x = (x1, ..., xd)T
Generate initial population of n fireflies xi (i = 1, 2, ..., n) using generation of
numbers with uniform distribution
Light intensity Ii at xi is determined by f(xi)
Define light absorption coefficient γ
Initial generation, k = 0

while (k < MaxGenerations)
  Update the generation number, k = k + 1
  for i = 1 to n (all n fireflies)
    for j = 1 to i (all n fireflies)
      if (Ij < Ii) in case of a minimization problem
        Move firefly i towards j in d-dimension;
      end if
      Attractiveness varies with distance r via equation (7)
      Evaluate new solutions and update light intensity
    end for j
  end for i
  Rank the fireflies and find the current best
end while
Postprocess results and visualization
```

Figure 1. Pseudocode of the FA.

A. Proposed FAC approach

The rate of convergence of evolutionary algorithms (EAs) is strongly influenced by the choice of certain parameters. For EAs in general, either one can manually tune the parameters before the run or control them during the run. The success of the search in EAs is highly dependent on a good balance between two processes: exploration and exploitation. Exploration allows searching the entire search space by ensuring the redirection of the search toward new regions, while exploitation favors a quick convergence toward the optimum.

In terms of this paper, the use of chaotic sequences in FA, a kind of EA, can be helpful to escape more easily from local minima than can be done through the traditional FA. Chaos is a kind of a feature of nonlinear dynamic system which exhibits bounded unstable dynamic behavior, ergodic, non-period behavior depended on initial condition and control parameters.

In this context, the proposed FAC uses chaotic sequences using Logistic map [20],[21] to tune α and γ given by constant values in the equation (7). The equation (7) is modified to

$$x_i = x_i + \beta_0 e^{-\gamma(t) \cdot r_{ij}^2}(x_j - x_i) + \alpha(t)\left(rand - \frac{1}{2}\right), \quad (8)$$

with

$$\gamma(t) = \mu_1 \cdot \gamma(t-1) \cdot [1 - \gamma(t-1)] \quad (9)$$

and

$$\alpha(t) = \mu_2 \cdot \gamma(t-1) \cdot [1 - \alpha(t-1)], \quad (10)$$

where t is the sample, and μ_1 and μ_2 are control parameters, $0 \le \mu_1, \mu_2 \le 4$. The behavior of the system of equations (9) and (10) is greatly changed with the variation of μ_1 and μ_2. The values of μ_1 and μ_2 determine whether $\gamma(t)$ and $\alpha(t)$ stabilize at constant sizes, oscillate between limited sequences of sizes, or behave chaotically in unpredictable patterns.

A very small difference in the initial value (t=1) of $\gamma(1)$ and $\alpha(1)$ causes substantial differences in its long-time behavior. Equations (9) and (10) are deterministic, displaying chaotic dynamics when $[\mu_1, \mu_2]$= 4 and $[\gamma(1), \alpha(1)] \notin \{0, 0.25, 0.50, 0.75, 1\}$. In this paper, $\gamma(t)$ and $\alpha(t)$ are adopted with chaotic dynamics and distributed in the range [0,1].

IV. SIMULATION RESULTS

Each individual of a population in tested FA and FAC approaches uses the variables vectors $\boldsymbol{n}$ and $\boldsymbol{r}$, where the boundaries are given in Section 2. During the evolution process, the integer variables n_i are treated as real variables; and in evaluating the objective function, the real values are transformed to the nearest integer values.

Each optimization method was implemented in MATLAB (MathWorks). All the programs were run under Windows XP on a 3.2 GHz Pentium IV processor with 2 GB of random access memory. To eliminate stochastic discrepancy, in each case study, it adopted 50 independent runs for each of the optimization methods involving 50 different initial trial solutions for each optimization method.

The total number of solution vectors in FA and FAC, i.e., the n was 15 fireflies and *MaxGenerations* = 200. The tested FA and FAC approaches adopt 3,000 objective function evaluations in each run. Furthermore, α= 0.6 and $\gamma = 1.0$ was adopted in FA.

In this work, the penalty-based method proposed in [13] was used in FA and FAC approaches for infeasible solutions (constraint violation). An approach is used to convert a constrained problem to an unconstrained one by modifying the search space. A penalty value is defined to take the constrained violation into account. The method proposed in [13] uses a procedure where the terms l are subtracted (maximization problem) from objective function $f(\boldsymbol{r},\boldsymbol{n})$ if $g(\boldsymbol{r},\boldsymbol{n}) > l$.

Table II shows the results over 50 independent runs for the overspeed protection system. FAC gives the best results, followed by the FA. By using the results in Table II, in terms of best $f(\boldsymbol{r},\boldsymbol{n})$ result, the solutions of FAC are just slightly better than the solution found by FA for the overspeed protection system.

The best result obtained for the overspeed protection system using FAC was 0.99995467, as shown in Table III. From Table IV, a best solution found by FAC for the overspeed protection system has a slight advantage in terms of solution quality (maximum value of $f(\boldsymbol{r},\boldsymbol{n})$) when compared with the results obtained by Chen [13], Dhingra [14], and Yokota *et al.* [15].

V. Conclusion

The study of flashing behavior and flash synchronization in fireflies [22]-[24] is a good inspiration to the development of bio-inspired optimization methods. In this context, the FA is a recently proposed metaheuristic algorithm inspired by the flashing behavior of fireflies.

This paper intended to provide a description of FA and a new FAC approach to a case study of reliability-redundancy optimization. Simulation results for a well-known benchmark problem in reliability-redundancy optimization presented in Tables II to IV reveal that FA and FAC demonstrate the effectiveness, efficiency, and robustness. Literature results are available for evaluated optimization examples in Table IV. FAC has a slight advantage in terms of solution quality over the other solvers.

As an extension of this paper, further investigation will be devoted to validate the proposed FAC in multiobjective applications of reliability-redundancy optimization.

Acknowledgments

The authors would like to thank National Council of Scientific and Technologic Development of Brazil - CNPq under Grant Nos. 568221/2008-7, 475689/2010-0, 302786/2008-2-PQ and 303963/2009-3/PQ for the financial support of this work.

References

[1] W. Kuo and V. R. Prasad, “An annotated overview of system-reliability optimization,” *IEEE Transactions on Reliability*, vol. 49, no. 2, pp. 176-187, 2000.

[2] W. Kuo, V. R. Prasad, F. Tillman and C. L. Hwang, Optimization Reliability Design: Fundamentals and Applications. Cambridge: Cambridge University Press, 2001.

[3] W. Kuo and R. Wan, “Recent advances in optimal reliability allocation,” *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 37, no. 2, pp. 143-156, 2007.

[4] M. Gen and Y. S. Yun, “Soft computing approach for reliability optimization: state-of-the-art survey,” *Reliability Engineering and System Safety*, vol. 91, no. 9, pp. 1008-1026, 2006.

[5] L. Jourdan, M. Basseur and E. G. Talbi, “Hybridizing exact methods and metaheuristics: a taxonomy,” *European Journal of Operational*, vol. 199, no. 3, pp. 620-629, 2009.

[6] E. G. Talbi, Metaheuristics: from design to implementation, Wiley, 2009.

[7] X. S. Yang, Nature-inspired metaheuristic algorithms, Luniver Press, 2008.

[8] X. S. Yang, “Firefly algorithms for multimodal optimization,” in *Stochastic Algorithms: Foundations and Applications* (SAGA 2009), Lecture Notes in Computer Sciences, vol. 5792, 2009, 169-178, 2009.

[9] M. S. Tavazoei and M. Haeri, “Comparison of different one-dimensional maps as chaotic search pattern in chaos optimization algorithms,” *Applied Mathematics and Computation*, vol. 187, no. 2, pp. 1076-1085, 2007.

[10] L. S. Coelho and V. C. Mariani, “Combining of chaotic differential evolution and quadratic programming for economic dispatch optimization with valve-point effect,” *IEEE Transactions on Power Systems*, vol. 21, no. 2, pp. 989-996, 2006.

[11] D. Yang, G. Li, and G. Cheng, “On the efficiency of chaos optimization algorithms for global optimization,” *Chaos, Solitons & Fractals*, vol. 34, no. 4, pp. 1366-1375, 2007.

[12] L. S. Coelho and R. B. Grebogi, “Chaotic synchronization using PID control combined with population based incremental learning algorithm,” *Expert Systems with Applications*, vol. 37, no. 7, pp. 5347-5352, 2010.

[13] T. C. Chen, “IAs based approach for reliability redundancy allocation problems,” *Applied Mathematics and Computation*, vol. 182, no. 2, pp. 1556-1567, 2006.

[14] A. K. Dhingra, “Optimal apportionment of reliability & redundancy in series systems under multiple objectives,” *IEEE Transactions on Reliability*, vol. 41, no. 4, pp. 576-582, 1992.

[15] T. Yokota, M. Gen and H. H. Li, “Genetic algorithm for nonlinear mixed-integer programming problems and it’s application,” *Computers and Industrial Engineering*, vol. 30, no. 4, pp. 905-917, 1996.

[16] S. Lukasik and S. Zak, “Firefly algorithm for continuous constrained optimization task,” in *International Conference on Computational Collective Intelligence* (ICCCI 2009), Lecture Notes in Artificial Intelligence, vol. 5796, 2009, 97-100.

[17] X. S. Yang, “Firefly algorithm, stochastic test functions and design optimization,” International Journal of Bio-inspired Computation, vol. 2, no. 2, pp. 78-84, 2010.

[18] X. S. Yang and S. Deb, Eagle strategy using Levy walk and firefly algorithms for stochastic optimization, in *Nature Inspired Cooperative Strategies for Optimization* (NISCO 2010), Studies in Computational Intelligence, Springer Berlin, vol. 284, pp. 101-111, 2010.

[19] M. K. Sayadi, R. Ramezanian and 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, no. 1, pp. 1-10, 2010.

[20] T. S. Parker and L. O. Chua, Practical numerical algorithms for chaotic system, Springer-Verlag, Berlin, Germany, 1989.

[21] S. H. Strogatz, Nonlinear dynamics and chaos, Perseus Publishing, Massachussetts, 2000.

[22] J. Buck and E. Buck, “Mechanism of rhythmic synchronous flashing of fireflies: fireflies of Southeast Asia may use anticipatory time-measuring in synchronizing their flashing,” *Science*, vol. 159, no. 281, pp. 1319-1327, 1968.

[23] D. Otte, “On theories of flash synchronization in fireflies,” *The American Naturalist*, vol. 116, no. 4, pp. 587-590, 1980.

[24] A. Moiseff and J. Copeland, “Firefly synchrony: a behavioral strategy to minimize visual clutter,” *Science*, vol. 329, no. 5988, pp. 181, 2010.

TABLE II

CONVERGENCE RESULTS OF $f(\boldsymbol{r},\boldsymbol{n})$ (50 RUNS) FOR THE OVERSPEED PROTECTION SYSTEM USING FA AND FAC APPROACHES

Optimization method	$f(\boldsymbol{r},\boldsymbol{n})$			
	Minimum (Worst)	Mean	Maximum (Best)	Standard Deviation
FA	0.99960531	0.99993121	0.99995466	0.00005086
FAC	0.99990212	0.99993907	0.99995467	0.00001447

TABLE III

BEST RESULT (50 RUNS) OF FA AND FAC APPROACHES FOR THE OVERSPEED PROTECTION SYSTEM

Parameter	**FA**	**FAC**
$f(\boldsymbol{r},\boldsymbol{n})$	0.99995466	0.99995467
n_1	5	5
n_2	6	5
n_3	4	4
n_4	5	6
r_1	0.90124442	0.90165488
r_2	0.85037025	0.88821801
r_3	0.94848940	0.94807430
r_4	0.88792842	0.84996263
MPI (%)	0.0225%	-
Slack (g_1)	55	55
Slack (g_2)	0.00000084	0.00934729
Slack (g_3)	24.80188272	15.36346308

Note: Slack is the unused resources.

$$MPI(\%) = [R_S(\text{FAC}) - R_S(\text{other})]/[1 - R_S(\text{other})]$$

TABLE IV

COMPARISON OF RESULT FOR THE OVERSPEED PROTECTION SYSTEM USING FAC WITH RESULTS IN THE LITERATURE

Parameter	**Dhingra [14]**	**Yokota *et al.* [15]**	**Chen [13]**	**This paper (using FAC)**
$f(\boldsymbol{r},\boldsymbol{n})$	0.99961	0.999468	0.999942	0.99995467
n_1	6	3	5	5
n_2	6	6	5	5
n_3	3	3	5	4
n_5	5	5	5	6
r_1	0.81604	0.965593	0.903800	0.90165488
r_2	0.80309	0.760592	0.874992	0.88821801
r_3	0.98364	0.972646	0.919898	0.94807430
r_4	0.80373	0.804660	0.890609	0.84996263
MPI (%)	88.6333%	91.6673%	23.5689%	-
Slack (g_1)	65	92	50	55
Slack (g_2)	0.064	-70.733576	0.002152	0.00934729
Slack (g_3)	4.348	127.583189	28.803701	15.36346308

Note: Slack is the unused resources.

$$MPI(\%) = [R_S(\text{FAC}) - R_S(\text{other})]/[1 - R_S(\text{other})]$$