20th IMEKO TC4 International Symposium and
18th International Workshop on ADC Modelling and Testing
Research on Electric and Electronic Measurement for the Economic Upturn
Benevento, Italy, September 15-17, 2014

Generalized extremal optimization of predictive maintenance to enhance monitoring of large experimental systems

P. Arpaia[1,2], M. Girone[1,2], D. Maisto[3], C. Manna[4], M. Pezzetti[2]

[1]*Department of Engineering, University of Sannio, Benevento, Italy, arpaia@unisannio.it*
[2]*European Organization for Nuclear Research (CERN), Geneva, Switzerland, mario.girone@cern.ch, marco.pezzetti@cern.ch*
[3]*Institute of High-Performance Computing and Networking, Naples, Italy, domenico.maisto@na.icar.cnr.it*
[4]*Cork Constraint Computation Centre, University College Cork, Cork City, Ireland, manna.cnr@gmail.com*

Abstract – **Predictive maintenance scheduling is an optimization problem aimed at defining the best activity sequence to minimize the expected cost over a time horizon. For very-large systems such as in experimental physics, maintenance optimization turns out to be very difficult owing to analytically intractable objective functions. In this paper, a meta-heuristic predictive maintenance algorithm based on the Generalized Extremal Optimization (GEO) is presented. With respect to state-of-the-art meta-heuristic techniques, the GEO-based maintenance algorithm allows optimization procedure to be configured easily through only one parameter without a numerous population. Preliminary results of the algorithm performance validation on the liquid helium storage system of the Large Hadron Collider at CERN are reported.**

I. INTRODUCTION

In the framework of large experimental systems [1], e.g. nuclear power plants, or transmission networks, the predictive maintenance scheduling problem is becoming more and more important. In such a context, the large number of components, often without simple interconnections, makes the problem analytically intractable. The maintenance optimization in multi-component systems is extensively studied in literature and components' interactions have been classified [2, 3]: structural, stochastic, and economic. Combining two or more models makes them too complicated to be analysed and for this reason, only the economic dependency is to be considered. Meta-heuristic techniques were extensively used for optimization problems and in particular the nature-inspired methods: Simulated Annealing (SA) [4], Genetic Algorithms (GA) [5], Particle Swarm Optimization (PSO) [6], and Ant Colony Optimization (ACO) [7]. A different meta-heuristic technique called Generalized Extremal Optimization (GEO) has also been applied for this class of problems. This method was developed by De Sousa and Ramos in 2002 [8] as a variation of the Extremal Optimization (EO) proposed by Boettcher adn Percus in 2001 [9], and since than has been applied to some complex optimization problems: e.g. De Sousa et al. applied GEO for an optimal heat pipe design [10] first, and later to an inverse radiative transfer problem [11]. Recently, in an effective multiprocessor scheduling [12], the above mentioned method was solved by Switalski and Seredynski. In all these works, GEO algorithm demonstrated satisfying results compared to the most popular algorithms.

Other versions of GEO's algorithm were also developed and applied for optimization problems, Guo et al. proposed a modified GEO (MGEO) for a quay crane scheduling problem [13]. The modification was introduced because of the various interference constraints imposed by this kind of problem. Also an hybrid GEO (HGEO) was proposed by D. Xie et al. that combines genetic and GEO algorithm [14]. They first developed a population-based GEO (PGEO) in order to accelerate convergence speed and than they integrated this new solution in HGEO showing better performances than classical GAs on an optimal power consumption for semi-track air-cushion vehicle.

In this paper, a standard GEO algorithm already proposed for predictive maintenance scheduling problems by the Authors [15] and compared to GA with encouraging preliminary results is extended to complex plants such as large experimental systems. In particular, in section II, the problem statement, including the mathematical formulation, is illustrated. In section III, the GEO algorithm is described and finally the experimental results of a case study on the CERN liquid helium storage system are reported in section IV.

II. PROBLEM STATEMENT

The main idea of this work is to evaluate GEO's algorithm performance on large experimental systems.

In this section, a function that takes into account the cost associated to the maintenance action (for example, the replacement or the calibration of a given component), and the cost associated to the system operating in the normal state (as monitoring, inspection and so on) is defined. Thanks to that, the effects of whatever given maintenance operation can be assessed.

Let N be the available resources to maintenance operation, and m_i (for i=1,..., M) the i-th system component that must be maintained (for a total of M components). The function C, representing the total cost of planned maintenance, can be expressed as:

$$C = \sum_{t=1}^{T}[\sum_{i \in G_t}(a_i + p_i(t) * B_i) + \sum_{j \in H_t}(k_j + b_j)] \quad (1)$$

where:

-T finite time horizon of planned maintenance;

-t for t = 1, ..., T, the t-th instant of the time horizon T;

-a_i the operating cost of the i-th component;

-k_j the replacement cost for the j-th component;

-b_j maintenance cost of the j-th component;

-$p_i(t)$ probability of failure of the i-th component at the time t;

-B_i cost of breakdown of the i-th component;

-G_t the set of every component not maintained at the time t;

-H_t the set of every maintained component at the time t.

Moreover, each given planned maintenance evaluated by means of (1) is subject to the following constraints:

i) Each m_i can be served (maintained) by only one of the N available resources at any time t;

ii) Each m_i has to be served at least one time instant t during the total time horizon T.

Finally, the probability of failure p_i at the time t could be derived from various deterioration models, depending on the type of monitored component, and from the nature of information or acquired signals.

III. THE PROPOSED METHOD

In the present work, each maintenance schedule S (called sequence, in the following) assessed by (1) is expressed through a binary string representation as:

$$S = [s_{11}, s_{12}, ..., s_{1M}; .....; s_{T1}, s_{T2}, ...s_{TM}] \quad (2)$$

where the single s_{TM} is the value of the corresponding bit. For example, s_{23} =1 means that the third component is maintained at the time instant t =2. The sequence representation in (2) is suitable for the proposed GEO approach. The maintenance problem is hard to solve even for apparently simple cases [19], as the time required for computing an optimal solution increases rapidly with the size of the study case.

A. *Generalized extremal optimization*

The goal of the proposed method is to find the best sequence, expressed as in (2), that minimizes the objective function (1) for the above problem. Let us consider a sequence (i.e., a maintenance schedule); a sequence can be encoded in a binary string, denoted by S of length (M*T) by means of the representation shown in (2). This sequence expression is particularly suitable to be faced through a GEO. Indeed, in analogy to what EO algorithm does, GEO works on a population (configuration) by muting, generation after generation, a single species (component) and by estimating the obtained candidate solution, for reaching the optimum. Thus, if each representation bit encodes a single species, then an entire population can be expressed by means of a binary string, hence by a sequence in the form (2) too. For the above reasons, a GEO algorithm can straightly work on a sequence S by evaluating the candidate solution to the considered maintenance problem through the cost function (1). This means that the lesser is the cost of the sequence the better is the scheduling. At each bit (species) is assigned a fitness value proportional to the decrease of the function (1) computed for the sequence with that bit flipped (i.e., mutated from 1 to 0 or vice versa). Then, each bit is ranked, such that: to the one with the least fitness is assigned rank 1, while to the one with the best fitness rank N. Subsequently, a new sequence is generated by flipping a bit chosen according the probability law:

$$P(k) \approx k^{-\tau}, 1 \leq k \leq N \quad (3)$$

where τ is a positive setting parameter.

A candidate solution in our GEO approach is a sequence S (assessed by (1)), composed of (M*T) bits, as defined in (2). An example of the GEO encoding consists of N design variables of 6 bits. Each bit is considered as a species [8]. In this example, M=6 components are maintained by N=3 resources in the time horizon T. This iterative process halts after a prefixed number of generation, and it returns the best sequence S_{best} which minimizes the objective function (1). The proposed procedure is described by the following pseudo-code:

1. Initialize a bit sequence S (with size M*T) randomly and evaluate the objective function C (as in (1));

Set: S_{best} = S and C_{best} = C(S);

2. For each generation:

a) For each bit i of S:

- Change the bit i (from 1 to 0, or vice versa) and evaluate the cost C(S_i)(as in (1))for S_i;

- Assess the fitness of bit i as ΔC(S_i)=C(S_i) - C_{best}
- Restore the bit i to its previous value.

b) Sort ΔC(S_i) in ascending way;

c) Choose the bit to change with probability (3);

d) Set S = S_i and C = C(S_i);

e) If C < C_{best} then set C_{best} = C, and S_{best} = S;

3. Return S_{best} and C_{best}.

As regard to the traditional evolutionary algorithms (GA, SA and so on), the present procedure has twofold advantages: (i) there is only one adjustable parameter τ, by simplifying the setting, and (ii) the entire evolution is made on one configuration solution S at the time, unlike the traditional evolutionary population-based algorithms, with lower computational costs and a better memory management.

IV. EXPERIMENTAL CASE STUDY

In this section, a case study on the liquid helium storage system installed at CERN is described and then some preliminary results of the proposed algorithm validation are presented. At this aim, the system configuration will be firstly described and then a comparison with GA will be reported, by analyzing the effectiveness (a measure of the quality solution within a given computational limit) and the efficiency (a measure of the amount of computing needed to achieve a satisfactory solution). The effectiveness is calculated as the best in run solutions, while the efficiency as the average of the iteration's number corresponding to the optimal solution.

A. *Helium storage system*

The cryogenic system of the Large Hadron Collider (LHC) under operation at CERN has a total helium inventory of 140 t [16] . Up to 50 t can be stored in gas storage tanks and the remaining inventory will be stored in a liquid helium storage system consisting of six 15 t of liquid helium tanks in 4 locations. In Fig.1, the general architecture of the system is shown.

The formulation proposed in this paper is aimed at solving the maintenance scheduling problem for only one of the LHC's installations, and for the sake of the simplicity, one its subset. A view of the 50 t liquid helium storage system located at LHC Point 1.8 is shown in Fig.2. It is composed by (i) two tanks, combined cryogenic line, which allows the transfer from the LHC to the tanks (saturated liquid helium) and back (gaseous helium at 5 K), (ii) bayonets to transfer helium from or to mobile containers to adjust the inventory when required [16], and (iii) a liquid nitrogen tank with a semi-rigid line, used to refill the thermal shield. All the low temperature parts are vacuum insulated and thermally protected by multilayer insulation.

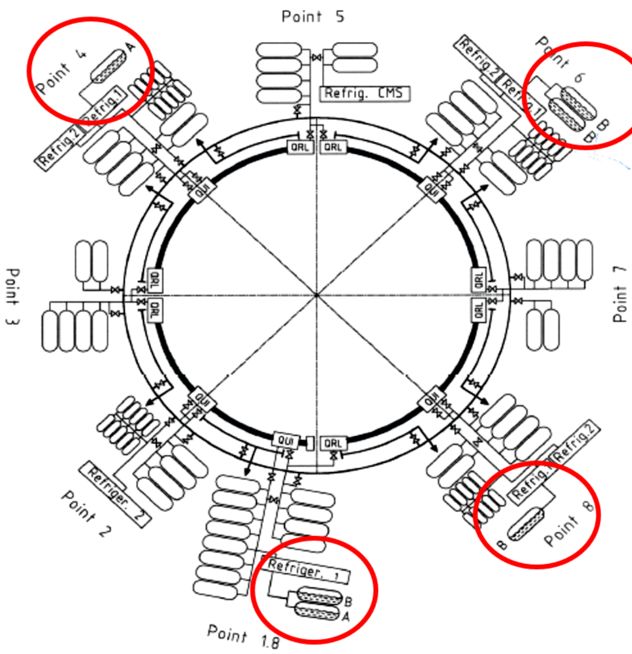

Fig. 1. Helium storage system architecture.

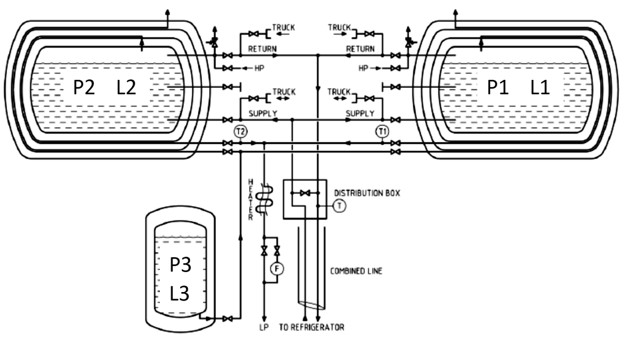

Fig. 2. Simplified scheme of LHC's point 1.8 storage system.

In Fig.2, each tank is equipped with a level meter (L1-L2), a pressure sensor (P1-P2), and a temperature sensor, installed on helium outlet shield line (T1-T2); as for the helium, the nitrogen tank is equipped with a pressure sensor (P3) and a level meter (L3); the distribution box is equipped with a thermometer (T) installed in the helium gas return line and the low-pressure line is equipped with a flow-meter (F) to measure the heat consumption of the tanks. All these sensors allow to manage the entire system and the measured values are sent to a dedicated PLC and a data acquisition system [17, 18].

The preventive maintenance scheduling for such a system is carried out by classifying all the components in different categories and by defining a task list including the correspondent costs for all the possible actions to carry out. As an example, mechanical elements are separated from sensors and from electrical equipment; for each category, all the possible components are identified and a list of tasks (inspection, replacement, cleaning, etc.) is drafted. The maintenance time horizon for each task is defined by experience. As a consequence, the drawback of this solution is that no intelligent scheduling for preventive maintenance is performed.

In this paper, an alternative solution to the above mentioned problem is presented exploiting the GEO algorithm in order to find the optimal sequence for maintenance operations which minimizes the costs.

B. System configuration

The system used to test the proposed methodology is the liquid helium storage described in section III.A and in particular the one represented in Fig.2. The components to be maintained can be classified in: control valves (CV), pneumatic valves (PV), analog sensors, digital sensors, hand valves (HV), safety valves (SV) and the breaking disks. For this case study, the GEO algorithm was implemented in Matlab and it was compared to a standard GA. For both the algorithms, a maximum number of iterations was fixed to 10^4. In GEO algorithm, the parameter τ was set to 0.75. As for GA algorithm, the following parameters configuration was used: number of runs: 10; population size: equal to the product M*T, crossover probability: 0.8; crossover mechanism: single point; mutation mechanism: uniform; and selection mechanism: roulette.

C. Experimental results

Scheduling problems with increasing dimensions were used to perform preliminary experimental tests in order to validate the effectiveness and the efficiency. In Table 1, the problem settings are presented.

Table 1. Problem Settings

Test n.	M	T	N	Solution length (n. of bits)
1	14	14	2	196
2	14	14	5	196
3	14	14	7	196
4	21	21	3	441
5	21	21	5	441
6	21	21	11	441
7	35	35	5	1225
8	35	35	10	1225
9	35	35	19	1225
10	42	42	5	1764
11	42	42	15	1764
12	42	42	23	1764

The proposed GEO works only on 1 individual and performs a number of evaluations as the solution length; therefore, in order to be compared with the GA, the population size of the last algorithm has been set equal to the solution length: 196 for the tests n.1-3, 441 for the test n.4-6, 1225 for the test n.7-9, and 1764 for the test n.10-12.

For a fixed number of maximum iterations, the GEO algorithm at each run converges to the same best solution cost with the same number of iterations. The same behaviour for the best solution in standard GA was observed, while a variable iterations number was noticed. In Tabs. 2 and 3, preliminary experimental results were reported.

Table 2. Best solution cost

Test n.	GEO Solution	GA Solution
1	784,964	–
2	805,404	–
3	802,484	784,964
4	1683,000	–
5	1716,701	–
6	1723,202	1650,240
7	4585,503	–
8	4597,201	–
9	4606,703	4326,38
10	6442,101	–
11	6561,102	–
12	6568,604	6137,244

Table 3. Number of iterations

Test n.	GEO Iter.	GA Iter. Mean	GA St. Dev.
1	3146	–	–
2	3483	–	–
3	9286	111,44	13,06
4	2963	–	–
5	8960	–	–
6	9218	234,33	19,55
7	4077	–	–
8	9763	–	–
9	9440	642,01	25,34
10	3995	–	–
11	9182	–	–
12	9517	960,50	31,78

For a small number of resources (N) in the time horizon (T), some problems to fulfil the constraints in GA algorithm were reported. The smallest value of N to avoid this problem, tests n.3, n.6, n.9, and n.12, is reported, but in these cases, the effectiveness (Table 2) achieved for GEO and GA are very different, and, in particular, GA performs better than GEO. N represents the number of available resources and, in practical applications, this value is always very limited. In Table 3, the efficiency expressed in terms of number of iterations is presented, and the difference between the GEO algorithm and GA is shown.

V. CONCLUSIONS

In the present paper, a standard Generalized Extremal Optimization (GEO) based algorithm for a predictive maintenance scheduling problem on large experimental systems has been presented. In these preliminary tests on a set of high dimension scheduling problems, an independence from N value for the GEO algorithm, compared to a standard GA, was highlighted. In particular, the proposed GEO is successful for each N value. However, by increasing N, the standard GA reaches better solutions with a lower number of iterations. Based on these results, a comparison with an evolutionary algorithm having the same feature, or new tests, implementing modified GEO algorithms, should be carried out in the future research.

REFERENCES

[1] D. Rehage, U.B. Carl, A. Vahl "Redundancy management of fault tolerant aircraft system architectures - reliability syntesis and analysis of degraded system states ", Aerosp. Sc. Technol., vol. 9, 2005, pp. 337-347.

[2] R. Dekker, R. Wildeman, F. van der Duyn Schouten, "A review of multicomponent maintenance model with economic dependence", Mathematical Methods of operations Research, vol.45, No.3, 1997, pp.411-435.

[3] R. P. Nicolai, R. Dekker, "Optimal maintenance of multi-component systems: a review", Component system Maintenance Handbook, London: Springer, 2008, pp. 263-286.

[4] S. Kirkpatrick, C. D. Gellat, M. P. Vecchi, "Optimization by Simulated Annealing", Science, vol. 220, No. 4598, 1983, pp. 671-680.

[5] D. E. Goldberg, "Genetic Algorithms in Search,Optimization, and Machine Learning", Addison-Wesley Publishing Company, 1989.

[6] J. Kennedy, R. C. Eberhart, "Particle Swarm Optimization", In Proc. of the 1995 IEEE International Conf. on Neur. Networks,IEEE Press, Piscataway, NJ, 1995, vol. 4, pp. 1942-1948.

[7] M. Dorigo, V. Maniezzo, A. Colorni, "The Ant System: Optimizing by a colony of cooperating agents", IEEE Trans. Syst. Man Cybern-Part B, 1996, vol. 26,No.1, pp 1-13

[8] F. De Sousa, M. R. Ramos, "Function Optimization using Extremal Dynamics", in 4th Int. Conf. On Inverse Problem in Engineering, Rio de Janeiro, Brazil, 2002.

[9] S. Boettcher, A. G. Percus, "Optimization with Extremal Dynamics", Physical Review Letters, 2001, vol. 86, pp. 5211-5214.

[10] F.L. De Sousa, V. V. Vlassov, F.M. Ramos, "Heat pipe design through generalized extremal optimization", Heat Transfer Engineering, 2004, vol. 25 (7),pp. 34-45.

[11] F.L. De Sousa, et al., "Application of the generalized extremal optimization algorithm to an inverse radiative transfer problem", Inverse Problems in Science and Engineering, 2007, vol. 15 (7),pp. 699-714.

[12] P. Switalski, F. Seredynski, "An effective multiprocessor scheduling with use of GEO metaheuristic", International Journal of Foundations of Computer Science, 2012, vol. 23 (2), pp. 465-481.

[13] D. Xie, L. Zhe, Y. Fan, "The computing of the optimal power consumption for semi-track air-cushion vehicle using hybrid generalized extremal optimization", Applied Mathematical Modelling, 2009, vol. 33, n.6, pp. 2831-2844.

[14] P. Guo, W.Cheng, Y. Wang, "A modified generalized extremal optimization algorithm for the quay crane scheduling problem with interference constraints", Engineering Optimization ahead-of-print,2013, pp. 1-19.

[15] P. Arpaia, D. Maisto, C. Manna, "A Generalized Extremal Optimization-inspired Algorithm for Predictive Maintenance Scheduling Problems", in IJCCI (ICEC), 2010, pp. 70-76.

[16] CERN Document, "The liquid helium storage system for the Large Hadron Collider", CERN 2010;

[17] CERN Report, "Control system", CERN

[18] CERN Document, "UNICOS a framework to built industry like control system: principles and methodology", CERN

[19] R. F. Stapelberg, "Handbook Reliability, Availability, Maintainability, and Safety in Engineering Design", Springer Verlag, 2009