Availability Optimization of Redundant System with Genetic Algorithm

Hélio Fiori de Castro
UNICAMP – State University of Campinas, Faculty of Mechanical Engineering, Department of Mechanical Design
Postal Box: 6051 - 13083-970 - Campinas, SP - Brazil
E-mail: heliofc@fem.unicamp.br

Katia Lucchesi Cavalca
UNICAMP –State University of Campinas, Faculty of Mechanical Engineering, Department of Mechanical Design
Postal Box: 6051 - 13083-970 - Campinas, SP - Brazil
E-mail: katia@fem.unicamp.br

Abstract. *In order to analyze the effect of the use of Genetic Algorithm in redundant system optimization problem, four availability optimization problems were considered, where the optimization variables were the number of redundancies of each component and the availability of the components. The difference between the problems is how the components availability is increased. In the first problem, the increase of the availability is reached with decrease of the repair time, which is represented by the MTTR (Mean Time to Repair). In the second case, the objective is the increase of the MTTF (Mean Time to Fail). The other problems object to get better repair time, as the first one. The difference between them is the way to reach this target. In the problem number three, a better MTTR is obtained with the increase of maintenance team. The other option (problem four) is optimize the maintenance resources, that is more general, because include maintenance team and other maintenance tools.*

Keywords*. Availability, Redundant System, Optimization, Genetic Algorithm.*

1. Introduction

The importance of designing reliable systems, which normally present high availability, is increasing, due to the engineering requirements of products with better quality and a higher safety level. There are two ways of increasing the availability of an engineering system: increase the availability of each component or use redundant components. In order to increase the availability of a component, it is possible to work on the improvement of reliability and maintainability. If the failure rate, which is related to the reliability, decreases, the system will be available to work for more time. And with an efficient maintenance program, the system can be repaired quickly. Reliability is the probability that a system or component operates successfully at an interval of time t. This success must be understood as the performance of the design function. The reliability can be easily obtained by failure time analysis of components or systems and it is complementary to the accumulated failure distribution. A statistical distribution must be used to represent the curves of reliability and accumulated failure distribution. The exponential distribution is characterized for a constant failure rate in the time domain, which is convenient for components or systems with long useful life, represented by electronic components. The Weibull distribution is ideal for fatigue failure and crack propagation, which are characteristics of mechanical systems. Other statistical distributions, such as normal, lognormal, Rayleigh and Gamma can be applied in failure analysis or maintenance analysis. Maintenance analysis has repair time as the control variable. It results in another probabilistic parameter, the maintainability, which is the capacity to renew a system or component in a determined period of time, to continue executing its design functions. The use of redundant components in an engineering system results in availability increase. However, the design and maintenance costs, along with volume and weight, also increase. So optimization methods are necessary to determine how many redundancies are necessary in each component or subsystem, maximizing availability while taking into account the constraint limits (cost, weight, volume). In the same way, the number of maintenance teams in each subsystem can be obtained. Traditional methods, such as the Lagrange Multiplier (Ramakumar, 1993), are inefficient with this kind of problem, because it is necessary to apply complex mathematical fundamentals, that makes the computational implementation difficult and without flexibility. Some search methods can reach only local optima. The Genetic Algorithm (Holland, 1975) is a search method that is analogous to biological evolution and reproduction. Castro and Cavalca (2002) applied some optimization methods to redundancy allocation problems. The Genetic Algorithm and Lagrange Multipliers were compared. Previous works show that Genetic Algorithm is indicated for problems, which apply complex mathematics expression for modeling. This kind of model hinders the use of differential calculus, which is basic for the traditional optimization method, such as the Lagrange Multiplier. Kumar et al. (1995), Painton and Campbell (1995), Rubinstein et al. (1997), Levitin (2001) Hsieh et al. (1998) have chosen the Genetic Algorithm to solve redundancy allocation problems and other reliability optimization problems. The aim of this paper is present an availability optimization problem, which is more complex than reliability problems, because availability considers maintenance and failure time. The optimization method is based on Genetic Algorithm, which is indicated for problems with this complexity.

2. Availability Optimization Problem

There are two possibilities of increasing the system availability. Firstly, it is possible to get high levels for the availability of each component, which can be obtained by the increase of failure time and/or the decrease of repair time. So, it is quite important to make the availability and dependability analysis of a product inside the decision process on economic and technical feasibility. Therefore, dependability is a probabilistic concept that relates failure time and repair time. Another way to increase the system availability is applying the concept of redundant components or subsystems. However, both ways of obtaining high availability levels bring high costs to the system. Redundant components must increase volume and weight as well. Therefore, optimization methods are necessary to obtain allowable costs, volume and weight at the same time as high availability levels. The availability optimization problem developed in this work uses the two possibilities to increase system availability. Because of that, some concepts of availability and dependability are presented in the following sections. Statistical distributions are needed to evaluate availability and dependability quantities. Thus, the exponential and Weibull distributions will also be explained. The choice of both distributions is due to the main focus of application of the methodology, which involves basically electronic and mechanical systems.

2.1 Availability Analysis

The steady state definition of Availability *A*, or the probability of an entity's successful operation in a determined period of time, can be calculated by the ratio between life time and total time between failures of the equipment. An entity is used to denote any component, subsystem, system or equipment that can be individually considered and tested separately.

$$A = \frac{life \quad time}{total \quad time} = \frac{life \quad time}{life \quad time + repair \quad time} \tag{1}$$

Life time is represented by *MTTF* (mean time to fail), which can be obtained from failure analysis. *MTTR* (mean time to repair) can be evaluated from maintenance analysis and represents the repair time as follows:

$$A = \frac{MTTF}{MTTF + MTTR} \tag{2}$$

Initially, to make the availability analysis, an exponential distribution is assumed to be representative for the reliability and maintainability statistical models. In this case, the *MTTF* is inversely proportional to the failure rate, which is constant (Equation 3). Analogously, the *MTTR* is the inverse of the repair rate (Equation 4).

$$MTTF = \frac{1}{\lambda} \tag{3}$$

$$MTTR = \frac{1}{\mu} \tag{4}$$

Availability can be expressed by the equation bellow:

$$A = \frac{\mu}{\mu + \lambda} \tag{5}$$

The time depended definition of Availability is given by Equation 6:

$$A(t) = \frac{\mu}{\mu + \lambda} + \frac{\mu}{\mu + \lambda} \cdot e^{-(\lambda + \mu)t} \tag{6}$$

If the time *t* tends to infinite:

$$\lim_{t \to \infty} A(t) = \frac{\mu}{\mu + \lambda} = A \tag{7}$$

2.2 Dependability Analysis

Dependability can be defined as the science of failures, which encompasses the knowledge of these failures, their assessment, their prediction, their measurement and their control. In the strict sense of the word, dependability is the ability of an entity to perform one or several required functions under given conditions.

The dependability concept was also defined by Wohl (1966) as the probability that an entity does not fail, or does fail and can be repaired in an acceptable period of time. This definition is an important design parameter, because it provides a single measurement of the performance conditions by means of the combination of the failure and repair rates associated to reliability and maintainability respectively.

An important characteristic of dependability is to allow the analysis of costs, reliability and maintainability simultaneously. Failure rate and repair rate are assumed as constant values for the dependability analysis, characterizing exponential distributions in both cases. The dependability ratio is the ratio between the repair rate μ, and failure rate λ.

$$d = \frac{\mu}{\lambda} = \frac{MTTF}{MTTR} \tag{8}$$

The higher the value of dependability ratio, the greater the necessity of maintenance in the system. So allied to the cost values, it is possible to determine if the equipment is economically and technically viable or not. The relation between the availability and the dependability ratio can be obtained from the combination of Equation 5 and Equation 8.

$$A = \frac{\mu}{\mu + \lambda} = \frac{\mu / \lambda}{1 + \mu / \lambda} = \frac{d}{1 + d} \tag{9}$$

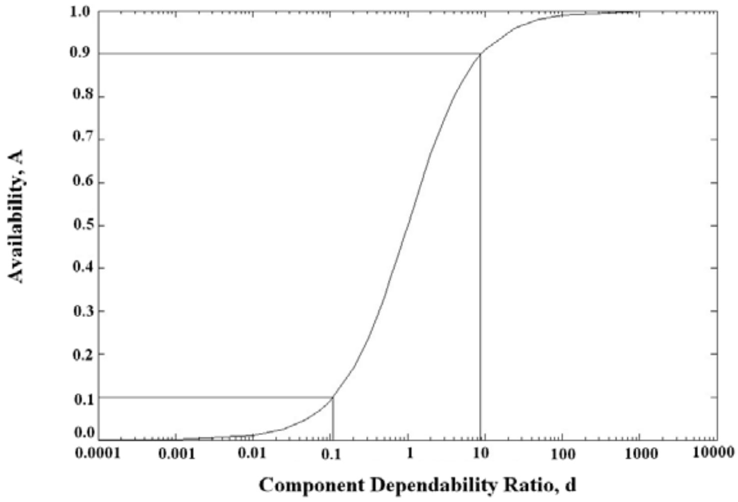

Figure 1. Relation between availability and dependability

Figure 1 shows a significant increase in the dependability ratio if the availability value is above 0.9 and a corresponding decrease if the availability value is less than 0.1 (Ertas, 1993). These effects mean great sensitivity of the dependability ratio in these regions. Analyzing the region where the availability is higher than 0.9, an extreme increase of maintenance is necessary for a small increase of the availability upgrade, which generates high costs. Therefore the region where the availability varies from 0.1 to 0.9 is indicated as economically and technically viable to work.

3. Problems Formulation

A redundant system can be represented by a series of parallel systems, as observed in Figure 2.

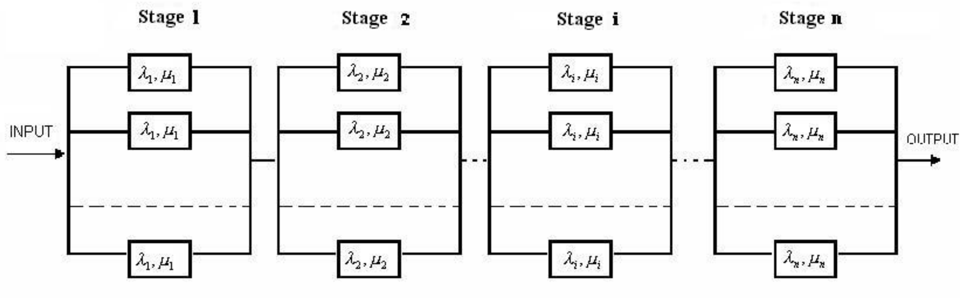

Figure 2. Redundant System

The availability of this system can be obtained by the Equation 10, where A_i is the availability of the components of the subsystem i and y_i is the number of redundant independent components in subsystem i.

$$A_S = \prod_{i=1}^{n} \left[1 - (1 - A_i)^{y_i}\right] \tag{10}$$

Considering an exponential distribution, the availability of each component A_i can be represented by Equation 9. Substituting for A_i in Equation 10, the availability function of the dependability ratio of each subsystem is obtained:

$$A_S = \prod_{i=1}^{n} \left[1 - \left(\frac{1}{1 + d_i}\right)^{y_i}\right] \tag{11}$$

The cost of the system can be obtained by the total sum of the product of each component cost by the number of components in that stage, as shown in Equation 12:

$$C = \sum_{i=1}^{n} c_i \cdot y_i \tag{12}$$

Similarly, the system weight and volume can be calculated:

$$W = \sum_{i=1}^{n} w_i \cdot y_i \text{ (a)} \quad V = \sum_{i=1}^{n} v_i \cdot y_i \text{ (b)} \tag{13}$$

In order to get higher levels of system availability, the number of redundant system y_i and the dependability ratio of a stage d_i have to increase. Four optimization problems were formulated to represent the increase of the dependability ratio.

3.1 Decrease of MTTR

The dependability ratio, given by Equation 6, can reach higher levels by the increase of the component *MTTF* or the decrease of its *MTTR*.

In order to get better *MTTR* levels, some maintenance investments are required. The investment cost can be represented by the following expression, where cd_i is the cost of dependability ratio increase.

$$C_M = \sum_{i=0}^{n} cd_i \cdot d_i \tag{14}$$

So the total cost is results from the sum of Equation 12 and Equation 14.

$$C_S = \sum_{i=1}^{n} (c_i \cdot y_i + cd_i \cdot d_i) \tag{15}$$

In that case, the optimization variable is the number of redundancies and the *MTTR* of each stage. The objective function is the availability, which is given by Equation 9. The constraints functions are:

1. Total Cost (Equation 15);
2. System Weight (Equation 13(a));
3. System Volume (Equation 13(b)).

3.2 Increase of MTTF

This problem is similar to the problem of section 3.1. The difference between then is the optimization variable. In the first problem the variable is the *MTTR* of each stage, and in the current problem it is the *MTTF* of each stage

3.3 Maintenance Team

Another way to decrease the *MTTR* and, consequently, get better dependability ratio d_i, is to consider the effect of the maintenance teams' action on the process. The influence of maintenance teams in cost and availability are formulated as the following equations.

The maintenance cost of the system can be obtained by Equation 16:

$$CM = \sum_{i=1}^{n} eq_i \cdot c_{eqi} + \sum_{i=1}^{n} q_i \cdot y_i \cdot c_{mi} \tag{16}$$

Where eq_i is the number of maintenance teams, y_i is the number of components in each stage, c_{eqi} is the maintenance team cost, c_{mi} is maintenance cost of the subsystem i and q_i is the failure probability of a component in subsystem i, which is given by Equation 17 for a exponential distribution.

$$q_i = 1 - e^{-\lambda_i \cdot t} \tag{17}$$

A subsystem fails only if all its components fail. Considering that only one team of maintenance works on a component and that the $MTTR_{1i}$ is the mean time to repair to the components of the subsystem i, the *MTTR* of all subsystem i can be obtained by Equation 18:

$$MTTR_i = f \cdot MTTR_{1i} \tag{18}$$

Where f is given by the expressions in Equation 19:

$$f = \begin{cases} \dfrac{y_i}{eq_i} & if \quad \dfrac{y_i}{eq_i} \in I \\ \text{int}\left(\dfrac{y_i}{eq_i}\right) + 1 & if \quad \dfrac{y_i}{eq_i} \notin I \end{cases} \tag{19}$$

The dependability ratio of the subsystem is given by Equation 20:

$$d_i = \frac{MTTF_i}{f \cdot MTTR_{1i}} = \frac{1}{f} \cdot d_{1i} \tag{20}$$

Where d_{1i} is the dependability ratio of one component in subsystem i.

Substituting d_i on Equation 9, the availability can be written as a function of the number of components and teams of maintenance in each subsystem.

$$A_S = \prod_{i=1}^{n} \left[1 - \left(\frac{1}{1 + \frac{1}{f} \cdot d_{1i}} \right)^{y_i} \right] \tag{21}$$

The objective is reaching the ideal number of components and teams of maintenance for the maximum value of the availability, inside the restriction area given by the following constrains:

1. Design Cost (Equation 12);
2. System Weight (Equation 13(a));
3. System Volume (Equation 13(b));
4. Maintenance Cost (Equation 16);
5. The number of components y_i must be higher than or equal to the number of teams of maintenance eq_i.

3.4 Maintenance Resources

A general way to describe the influence of all kind of maintenance resources on availability optimization problem is formulated at this section. Maintenance resources are represented by maintenance teams, equipments, financial resources and other kind of engineering concepts that can be applied to get better *MTTR* value.

In order to describe the influence of maintenance resources on availability value, it is defined the impact variable I, that is how much the *MTTR* is decreased if 100% of maintenance resources is applied on any component. The dependability ratio d_i is given by:

$$d_i = \left[(I-1) \cdot rec_i + 1 \right] \cdot d_{1i} \tag{22}$$

Where rec_i is the amount of maintenance resources that is applied on stage i, and d_{1i} is the dependability ratio in the case that no maintenance resources is applied.

The maintenance cost is given by Equation 23. And the system availability, which is the objective function, can be getting by the combination of Equation 11 and Equation 22.

$$CM = \sum_{i=1}^{n} rec_i \cdot crec_i + \sum_{i=1}^{n} q_i \cdot y_i \cdot cm_i \tag{23}$$

The constrained functions are:

1. Design Cost (Equation 12);
2. System Weight (Equation 13(a));
3. System Volume (Equation 13(b));
4. Maintenance Cost (Equation 23);

4. Genetic Algorithm

The search space, determined by the restrictive conditions, and the objective function, are the only parameters necessary for some search algorithms such as the Evolution Strategy (Michalevicz, 1996) and Genetic Algorithm (Mitchell, 1996). A genetic algorithm is a search strategy that employs random choice to guide a highly exploitative search, striking a balance between exploration of the feasible domain and exploitation of "good" solutions (Holland, 1992). This strategy is analogous to biological evolution. Different to the classic optimization algorithms, the Genetic Algorithm (GA) does not work with only one point in the search space, but with a group of points simultaneously. The number of points is previously determined by a parameter known as population size. The GA does not need to use differential calculus. It can be considered a robust method, because it is not influenced by local maximum or minimum, discontinuity or noise in the objective function. The GA applied to the reliability optimization had a considerable increase in the 80's and 90's, which is shown by the evaluation of Kuo and Prasad (IEEE Transactions on Reliability, 2000). Another example of this application is the work of Hisieh, Chen, Bricker (Microelectronics Reliability, 1998). In this work, the redundant number and the reliability of the components of each subsystem (stage) are optimized for three different systems.

4.1 Genetic Algorithm operators

The GA operators are the instruments used by the algorithm to reach the optimum point of the function (Goldberg, 1989). Three operators were developed in the program: mutation, crossover and selection. The mutation is the operator of the GA that changes some characters of the selected chromosomes, forming a new individual. Crossover is an operator that mixes the "genotype" of two selected chromosomes. Figure 3 illustrates mutation and crossover process.

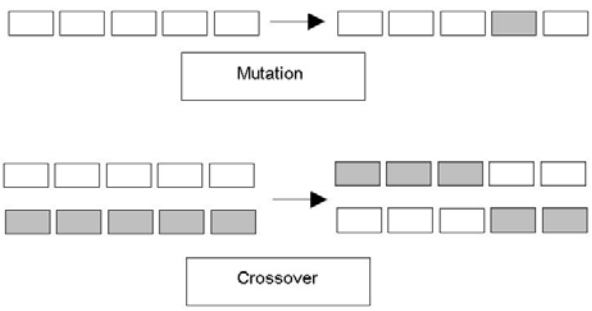

Figure 3. Mutation and Crossover examples

Inversion is an operator that inverts the order of the "genes" of an individual, forming a new individual with a different genetic code. The other operator is selection, which selects the fitter individuals (Objective Function closer to the optimum point), in order to be the genitors of the next generation.

4.2 Genetic Algorithm codification

Binary numbers traditionally represents a genetic algorithm individual. It makes working with integer and real numbers together in the same optimization process possible. Therefore, decoding transforms this variable in binary numbers. However, it is possible to use different kind of codes, such as genes that are represented by integer and real numbers.

The decoding of a binary sequence to decimal number (integer or real) is represented by Equation 24:

$$x_j = c_j + \sum_{i=0}^{k-1} b_i \cdot 2^i \cdot \frac{d_j - c_j}{2^k - 1} \tag{24}$$

Where c_j and d_j are the maximum and minimum possible values of the decimal variable x_j and b_i is the digit i of a binary number with k digits. Thus, the number of digits of an individual (chromosome) is the product of the number of variables (redundancy and maintenance team) and the number of bits (eight).

4.3 Genetic Algorithm parameters

There are four GA parameters that influence the process time and the objective function convergence. As the GA is characterized to be a search algorithm, the increase of the operation time brings about better objective function convergence. The GA parameters are:

- Total number of generations: this parameter is characterized to be the stop condition of the genetic algorithm. The increase of the total number of generations results in a linear increase of the process time as shown on Figure 4;
- Population size: it is the number of individuals, who are represented by their chromosomes in each generation. The increase of this parameter increases the probability of objective function convergence. However, the process time increases very significantly. This influence is represented by Figure 5

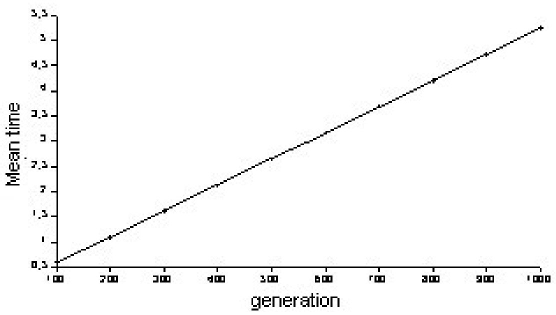

Figure 4. Influence of total number of generations on process time

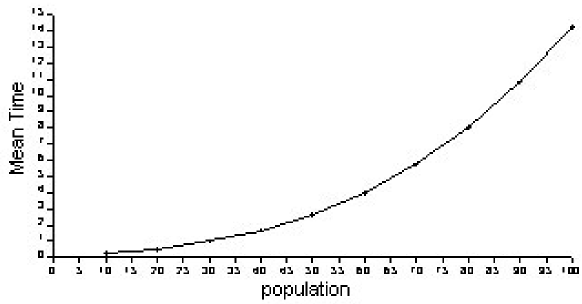

Figure 5. Population influence on process time

- Mutation probability: it is the probability of mutation occurrence. Normally, the increase of mutation probability leads to better values of availability. Over 90% mutation probability, this influence is negligible and no improvement is noticed in the availability value. Therefore, in this case-study, a mutation probability of 90% was considered.
- Crossover Probability: it is the probability of the crossover occurrence. The increase of crossover probability leads to better values of availability up to the value of 10%. If no crossover is applied, the process does not reach the best optimum result. For a crossover probability over 10% no significant improvement is noticed in the availability value and its value tends to decrease. Therefore, in this case-study, a crossover probability of 10% was considered.
- Inversion Probability: it is the probability of the inversion occurrence.

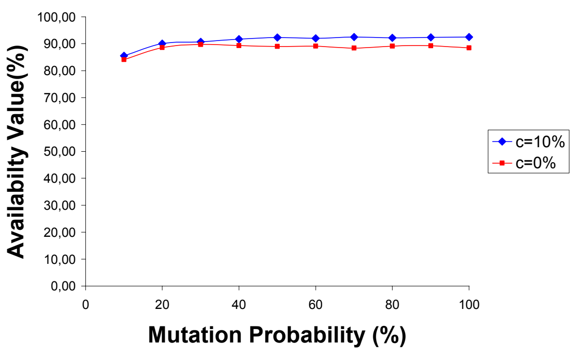

Figure 6. Mutation influence on the Availability Value

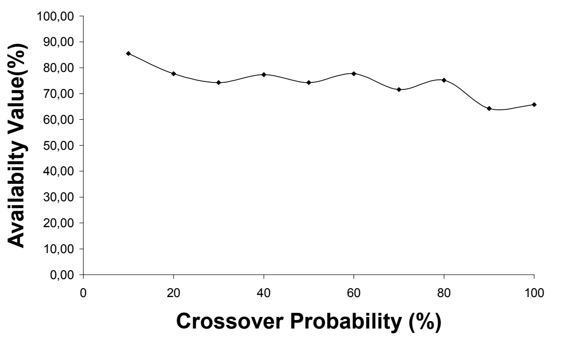

Figure 7. Crossover influence on the Availability Value

4.4 Genetic Algorithm Steps

1. Form an initial generation;

2. Make generation = 1;
3. Select 50% of the best individuals (closest to the optimum point);
4. Form new individuals using mutation and crossover, until the population size is reached;
5. Make generation = generation + 1;
6. If generation is equal to the total number of generations then stop, otherwise go to step 3.

5. Numerical Simulation

In order to analyze the results of the proposed problems, a system with five subsystems is chosen. The MTTF, MTTR, weight and volume of each component are in Table 1. All cost variables are shown Table 2:

Table 1. System Data.

Subsystem	MTTF	MTTR	WEIGHT	VOLUME
1	500	50	50	55
2	550	35	45	50
3	600	40	80	70
4	750	30	35	35
5	500	30	70	65

Table 2. System data (Cost).

Subsystem	Design Cost	Dependability Ratio increase Cost *(Problems 1 and 2)*	Maintenance Cost *(Problems 3 and 4)*	Team Maintenance Cost *(Problem 3)*	Maintenance Resource Cost *(Problem 4)*
1	50	0,60	60	30	5
2	55	0,60	40	30	5
3	55	0,50	45	25	3
4	40	0,50	30	25	3
5	60	0,60	50	30	5

The constraints considered to be the problem here were: maximum design and maintenance costs, weight of the system, volume of the system, and number of available maintenance teams for the system. The number of maintenance teams in each subsystem cannot be larger than the number of redundancies. Table 3 shows the constraint values for the problem.

Table 3. System Constraints.

Maximum Design cost	Maximum Weight	Maximum Volume	Maximum Maintenance Cost	Available maintenance team
1000	1000	1000	500	20

The Genetic algorithm parameters for this simulation are:

Table 4. Genetic Algorithm Parameters

	Total Number of generations	Population size	Probability of mutation	Probability of crossover	Probability of inversion
Problem 1	5000	50	90%	10%	70%
Problem 2	5000	50	90%	10%	70%
Problem 3	10000	50	90%	10%	70%
Problem 4	5000	50	90%	10%	70%

In order to get the maintenance cost in problems 3 and 4, the value of the accumulative failure distribution in the time domain is necessary. So time t is considered equal to 100 time unit (the same unit as MTTF and MTTR). The optimum result of problem one can be observed in Table 5 and Table 6 show the best result for problem 2.

Table 5 – Optimum Solution of problem 1

Subsystem	Redundancies	MTTR
1	2	4,18
2	3	4,05
3	2	3,59
4	2	4,25
5	3	7,67

Table 6. Optimum Solution of problem 2

Subsystem	Redundancies	MTTF
1	3	2827
2	2	5000
3	3	1227
4	4	699
5	4	1227

Table 7. Optimum Solution of problem 3.

Subsystem	Redundancies	Maintenance Team
1	3	3
2	4	2
3	4	3
4	4	2
5	3	3

Table 8. Optimum Solution of problem 4.

Subsystem	Redundancies	Maintenance Resources
1	3	22%
2	3	15%
3	3	38%
4	6	1%
5	3	15%

The optimum solution for the simulation of problem 3 is shown in Table 7. The best result of problem 4 is shown in Table 8. It is noted that is used only 91% of maintenance resources. The other 9% is not used because it causes a break in restrictions conditions. So, is not viable to use all maintenance resources.

The system characteristics for the optimum solution are:

Table 9. Final Systems

	Problem			
	1	2	3	4
Optimum availability for the system	99,99%	99,99%	99,87%	99,98%
Design Cost	$998,8	$825	$930	$900
System Weight	675	900	1000	945
System Volume	625	875	980	930
Maintenance Cost	-	-	$494	$500
Number of maintenance Teams	-	-	13	-

All system characteristics are near to the restriction conditions. Others constraints were considered to simulate the problem 3, as shown in Table 10:

Table 10. System Constraints

Maximum Design cost	Maximum Weight	Maximum Volume	Maximum Maintenance Cost	Available maintenance team
1500	1500	1500	1000	20

The optimization results for this second case study are listed in Table 11 and the new system characteristics are shown in Table 12:

Table 11. Optimum solution for new constraints

Subsystem	Redundancies	Maintenance Team
1	7	4
2	4	4
3	6	3
4	4	4
5	5	5

Table 12. Final System for new constraints

Optimum availability for the system	100 %
Design Cost	$1360
System Weight	1500
System Volume	1470
Maintenance Cost	$769.5
Number of maintenance Teams	20

In this second case, where the search space is larger, the optimum result (availability equal to 100%) can be obtained through many combinations of the parameter response. Therefore, a multiobjective optimization would be advisable, and besides maximizing the availability, it would minimize the costs, in order to improve the convergence of the algorithm.

6. Real Example

In order to show the viability of the proposed method, a real example of a production line of ZF do Brazil S.A. was adopted for problem 3. The data for this analysis is shown in Table 13. The system constraints can be observed in Table 14.

Table 13. Real Example data

Subsystem	MTTF	MTTR	Area	Design Cost	Maintenance Cost	Team Maintenance Cost
1	7670	42	15	500000	200	22,28
2	7670	42	15	500000	200	22,28
3	5200	120	20	700000	300	22,28
4	5700	160	15	200000	100	22,28

Table 14. System Constraints

Maximum Design cost	Maximum Area	Maximum Maintenance Cost	Available maintenance team
5000000	1000	3000	10

Area can be calculated similarly to the volume. Volume and weight are not considered in this case. The total number of generation is 5000 generations. The other Genetic Algorithm parameters are the same of Table 4. The optimization results for this case and the system characteristics are shown in Tables 15 and 16 respectively.

Table 15. Optimum solution for real Example

Subsystem	Redundancies	Maintenance Team
1	1	1
2	1	1
3	2	2
4	2	2

Table 16. Final System for real Example

Optimum availability for the system	98,79 %
Design Cost	$2800000
System Area	100
Maintenance Cost	$169,6
Number of maintenance Teams	6

7. Conclusions

The Genetic Algorithm solves the proposed problems, but if the GA parameters (total number of generation, population size and probability of mutation, crossover and inversion) are not properly adjusted, the result can be unsatisfactory. Thus, a wide knowledge of the analyzed system and of the genetic algorithm optimization method is necessary. The more units the system has, the larger the total number of generations and population size. In this case, the total number of generations increases, in order to reach the best result. For problems with large search space, the possibility of reaching availability value close to 100% with many combinations increases. In that case, the search space needs to be diminished or a multi-objective operator is necessary. The real application proved that the proposed method is viable for practical using in engineering.

8. Acknowledgements

The authors would like to thank CNPq for the financial support for this work, as well as Eng. Alexandre Degani and ZF do Brasil S.A. for the practical data for this work.

9. References

Castro, H. F., Cavalca, K. L. Reliability Optimization of Redundant System. SAE Technical Papers, Danvers MA USA, v. 1, n. 1, p. 1-9, 2002.

Doty, L. A. Reliability for the technologies. New York: ASQC Quality press Book. 1989, 307p.

Ertas A, The Engineering Design Process, John Wiley & Sons Inc, 1993, 525p.

Goldberg, D. E. Genetic Algorithm in search optimization and Machine Learning. Reading, Mass.: Addison-Wesley, 1989, 412p.

Hsieh, Y. C., Chen, T. C., Bricker, D. L. Genetic Algorithms for Reliability Design Problems. Microelectronics Reliability, v. 38, n. 10, pp. 1599-1605, 1998.

Holland, J. H. Adaptation in natural and artificial system: an introduction with application to biology, control and artificial intelligence, An Arbor, MI: University of Michigan Press, 1975.

Holland, J. H. Adaptation in natural and artificial system: an introduction with application to biology, control and artificial intelligence, 2nd. Ed. Cambridge, MA: MIT Press, 1992, 211p.

Kuo, W., Prassad, V. R. An Annotated Overview of System Reliability Optimization. IEEE Transactions on Reliability, v. 49, n. 2, pp.176-187, June 2000.

Kumar, A., Pathak, R. M., Gupta, Y. P. Genetic-Algorithm-Based Reliability Optimization for Computer Network Expansion. IEEE Transactions on Reliability, v. 44, n. 1, pp. 63-72, March 1995.

Levitin, G. Redundancy Optimization for Multi-state System with Fixed Resource-Requirements and Unreliable Sources. IEEE Transactions on Reliability, v. 50, n. 1, pp. 52-59, March 2001.

Lewis, E. E. Introduction to Reliability engineering. John Wiley & Sons, Inc. 1996, 435p.

Michalewicz, Z. Genetic Algorithms + data structures = evolution programs, 3rd. Ed. New York: Springer, 1996, 387p.

Mitchell, M. An Introduction to Genetic Algorithm. Cambridge: MIT, 1996, 209p.

Painton, L., Campbell, J. Genetic Algorithm in Optimization of System Reliability. IEEE Transactions on Reliability, v. 44, n. 2, pp. 172-178, June 1995.

Ramakumar, R., Engineering Reliability, Ed. Prentice Hall, 1993, 482p.

Rubinstein, R. Y., Levitin, G., Lisnianski, A., Ben-Haim, H. Redundancy Optimization of Static Series-Parallel Reliability Models Under Uncertainty. IEEE Transactions on Reliability, v. 46, n. 4, pp. 503-511, December 1997.

Villemeur, A. Reliability, availability, maintainability and safety assessment – Methods and Techniques. Chichester, UK: John Wiley & Sons, 1992, 365p.

Wohl, J. G. System Operational Readiness and Equipment Dependability. IEEE Transactions on Reliability, v. R-15, n. 1, pp. 1-6, May 1966.