

Optimal Design of Reliable Network Systems in Presence of Uncertainty

Marzio Marseguerra, Enrico Zio, Luca Podofillini, and David W. Coit, *Member, IEEE*

Abstract—In practice, network designs can be based on multiple choices of redundant configurations, and different available components which can be used to form links. More specifically, the reliability of a network system can be improved through redundancy allocation, or for a fixed network topology, by selection of highly reliable links between node pairs, yet with limited overall budgets, and other constraints as well. The choice of a preferred network system design requires the estimation of its reliability. However, the uncertainty associated with such estimates must also be considered in the decision process. Indeed, network system reliability is generally estimated from estimates of the reliability of lower-level components (nodes & links) affected by uncertainties. The propagation of the estimation uncertainty from the components degrades the accuracy of the system reliability estimation. This paper formulates a multiple-objective optimization approach aimed at maximizing the network reliability estimate, and minimizing its associated variance when component types, with uncertain reliability, and redundancy levels are the decision variables. In the proposed approach, Genetic Algorithms (GA) and Monte Carlo (MC) simulation are effectively combined to identify optimal network designs with respect to the stated objectives. A set of Pareto optimal solutions are obtained so that the decision-makers have the flexibility to choose the compromised solution which best satisfies their risk profiles. Sample networks are solved in the paper using the proposed approach. The results indicate that significantly different designs are obtained when the formulation incorporates estimation uncertainty into the optimal design problem objectives.

Index Terms—Estimation uncertainties, genetic algorithms, Monte Carlo simulation, multi-criteria optimization, network reliability optimization.

ACRONYMS¹ AND NOMENCLATURE

GA	Genetic Algorithms
MC	Monte Carlo
\mathbf{x}	vector of the control variables of the optimization (see below)
$f(\mathbf{x})$	value of the objective function (fitness) in correspondence of the vector in the single objective case

Manuscript received June 17, 2002; revised October 3, 2004. Associate Editor: W. H. Sanders.

M. Marseguerra, E. Zio, and L. Podofillini are with the Department of Nuclear Engineering, Polytechnic of Milan, 20133 Milan, Italy.

D. W. Coit is with the Department of Industrial & Systems Engineering, Rutgers University, Piscataway, NJ, USA.

Digital Object Identifier 10.1109/TR.2005.847279

¹The singular and plural of an acronym are always spelled the same.

N	number of objective functions in the multi-objective case
$f_h(\mathbf{x})$	value of the h -th objective function (fitness) in correspondence of the vector in the single objective case, $h = 1, 2, \dots, N$
N_{ga}	number of individuals in the GA population
s	number of total sections (node pairs or links) constituting the network
m_i	number of component types available for node pair i or link i , $i = 1, 2, \dots, s$
N_k	total number of available component types, $N_k = \sum_{i=1}^s m_i$
x_{ij}	number of components of type j used in node pair i or link i , $i = 1, 2, \dots, s$, $j = 1, 2, \dots, m_i$.
c_{ij}	cost of component of type j used in node pair i or link i , $i = 1, 2, \dots, s$, $j = 1, 2, \dots, m_i$.
w_{ij}	weight of component of type j used in node pair i or link i , $i = 1, 2, \dots, s$, $j = 1, 2, \dots, m_i$.
$C(W)$	maximum network cost (weight) allowed by design
$N_C = \sum_{i=1}^s \sum_{j=1}^{m_i} x_{ij}$	number of components constituting the network
$R(\mathbf{x})$	network reliability, for design vector \mathbf{x} , $\mathbf{x} = \{x_{11}, x_{12}, \dots, x_{21}, x_{22}, \dots, x_{n1}, x_{n2}, \dots, x_{nm_n}\}$
q_k	unreliability of a component of type k , $k = 1, 2, \dots, N_k$,
n_t^k	number of reliability tests for component of type k , $k = 1, 2, \dots, N_k$,
$Var(\hat{R}(\mathbf{x}))$	variance of the network reliability estimate, $\hat{R}(\mathbf{x})$, for design vector \mathbf{x}
$E(\bullet), Var(\bullet), \hat{\bullet}, \tilde{\bullet}$	expected value, variance, estimate, realization of random variable \bullet
N_s	number of network life histories performed with the same set of sample reliability estimates
N_R	number of different sets of sample reliability estimates considered for uncertainty propagation
N_f	number of systems failures observed over N_s life histories

I. INTRODUCTION

SYSTEM network design problems are becoming increasingly critical & complex as telecommunication networks (and others) are expanded & upgraded in response to consumers' information needs.

The design & operation of a network with high reliability requirements are difficult, yet essential tasks. Two approaches are often used to improve the reliability of a network system. One is to allocate redundant components with similar functionality between two nodes. The allocation of redundancy, however, implies additional cost, weight, and maintenance requirements on the entire network system. The second approach is to implement a single highly reliable component between a pair of nodes, perhaps as a replacement for an existing, less reliable link. The associated disadvantage is that highly reliable components are usually more expensive, but the overall system weight, complexity, and maintenance costs may be reduced compared to resorting to redundancy allocation.

Furthermore, practical network designs have multiple objectives. In addition to high system reliability, there are other critical objectives such as low implementation costs, low risks of damages associated with system failures, etc. These objectives are usually conflicting (e.g., low cost vs. high reliability) so that the final design is always a compromised solution.

There has been much significant research considering network reliability optimization. Kiu & McAllister [1] reported that the assignment of the proper reliability level to the links of a network with fixed topology, as a means to optimize the system reliability, is an NP-hard problem. The approach proposed was that of assigning the proper reliability to node pairs, so as to obtain near maximal reliability based on the importance order of the links. Altıparmak *et al.* [2] used Genetic Algorithms (GA) to determine the optimal design configuration by selecting components from multiple alternatives. Several other examples of methods & applications have been proposed. In general, in these previous research efforts, the focus is on single objective problems, e.g., maximization of the network reliability.

In practice, it is often insufficient to simply maximize the estimate of the system network reliability. In particular, decision-makers want to consider the uncertainty in the reliability estimation so they can avoid 'risky' solutions characterized by unsatisfactorily high estimation uncertainty. Thus, the problem is often inherently multi-objective.

Indeed, the estimation of the reliability of a network is based on the knowledge of the network topology, and on the estimates of the reliability of its components. However, the components' reliability estimates are often uncertain, particularly for those components for which only sparse reliability data are available. Thus, the resulting system reliability estimate suffers from uncertainty as well.

System designers & users are generally risk-averse. They generally prefer a network design with a high reliability estimate assured by a low estimation uncertainty. Thus, maximization of the system reliability, and minimization of its estimation uncertainty, become mutually important, although the latter is hardly ever emphasized. The variance of the system reliability estimate, for example, could be regarded as a significant indicator of the estimation uncertainty at the system level.

Several papers have been published to address redundancy allocation of series-parallel systems under multiple objectives. Dhingra [3] applied a multiple-criteria optimization approach to maximize system reliability, and minimize resource consumption (cost, weight and volume). Misra & Sharma [4], [5] used integer programming, and a min-max approach to obtain Pareto optimal solutions. More recently, Busacca *et al.* [6] applied GA to find a set of Pareto optimal solutions with application to nuclear safety systems. For network system design, there have been only limited research considering multiple objectives, and uncertainty in the estimate has not been considered within a multi-objective framework.

Reliability optimization research has rarely considered the uncertainty in the reliability estimation; and for network systems, this remains an under-developed research area. Rubinstein *et al.* [7] use a GA to maximize the expectation of a series-parallel system reliability estimate through component redundancy allocation with component uncertain properties. Maximization of the expectation of the reliability estimate may not suffice in many practical cases, if it is important to avoid system designs with an unacceptable degree of uncertainty. Thus, it is desirable to develop a multiple-objective optimization algorithm for network systems to select an optimal design strategy which explicitly considers the component uncertainty.

In this paper, multi-objective GA are utilized to select optimal network designs that balance the dual objectives of high reliability, and low uncertainty in its estimation. This new approach exploits the Monte Carlo (MC) simulation as the objective function evaluation engine, and Pareto optimality as the GA preference criteria.

GA are numerical search tools which operate according to procedures that resemble the principles of natural selection & genetics [8], [9]. Because of their flexibility and widespread applicability, GA have been successfully used in a wide variety of problems in several areas of engineering, and life science [10]–[15]. In particular, the application of genetic, and other evolutionary algorithms, to multi-objective optimization problems is subject to great attention in the technical community, as demonstrated by many recent publications in the field [6], [16]–[20].

The utilization of GA for the multi-objective optimization problem entails comparing two solutions with respect to the multiple objectives considered [6], [21], [22]. In the case of a single-objective, the comparison is trivial because a vector solution \mathbf{x} is better than \mathbf{y} if the corresponding objective function value $f(\mathbf{x})$ is greater than $f(\mathbf{y})$. For GA, the objective function is defined as the *fitness*. The fitness provides a metric to evaluate & rank each prospective solution, and also the fitness is used as criteria to select parents to form the next generation. When the situation is extended to N objectives, there are N objective functions, $f_h(\cdot)$, $h = 1, 2, \dots, N$. Because it is unlikely that a single solution will maximize all N objective functions, it is necessary to define criteria to compare & rank solutions. Two solutions \mathbf{x} & \mathbf{y} can be compared in terms of the *dominance* of one solution over the other with respect to all N objectives. As a result of the multi-objective search process, convergence is achieved on a Pareto-optimal set of nondominated solutions. These Pareto optimal solutions are generally sub-optimal if considered one at a time.

II. PROBLEM FORMULATION

The decision variables of the problem of interest are the numbers of components, x_{ij} , of a given type j to be allocated in the various sections (node pairs & links) i of a network system, $i = 1, 2, \dots, s$, and $j = 1, 2, \dots, m_i$. The network is designed to maximize two objectives: the expectation of the network reliability, and the negative of the variance of such estimate. This is a sound formulation for a risk-averse decision maker who may prefer a network design with a marginally lower expected value of reliability, rather than one with a higher expected value, provided that the estimate of the former is known to be less uncertain. In practice, many decision makers prefer a risk-averse solution, whereas most optimization algorithms require or assume risk-neutrality.

Introducing cost & weight constraints on the feasible solutions, the multi-objective optimization problem may be formulated as follows:

Problem P

$$\max \left\{ E \left[\hat{R}(\mathbf{x}) \right], -Var \left(\hat{R}(\mathbf{x}) \right) \right\}$$

subject to:

$$\sum_{i=1}^s \sum_{j=1}^{m_i} c_{ij} x_{ij} \leq C$$

$$\sum_{i=1}^s \sum_{j=1}^{m_i} w_{ij} x_{ij} \leq W$$

III. MONTE CARLO APPROACH FOR ESTIMATING THE RELIABILITY OF A NETWORK, AND ITS ASSOCIATED VARIANCE

The computational complexity involved in the estimation of the reliability of a network increases significantly as the network dimension increases. For large, complex networks, different analytical estimation techniques have been attempted in the past, inevitably resorting to simplifications, sometimes unrealistic, in the system models. Thanks to the increasing computational power available, other techniques, traditionally considered too time consuming, are now becoming practical. In particular, Monte Carlo (MC) simulation provides a useful tool for achieving a realistic estimate of network reliability [23].

The estimation of the reliability of a network is obtained on the basis of the reliability estimates of the N_C components, $N_C = \sum_{i=1}^s \sum_{j=1}^{m_i} x_{ij}$, which form the constitutive parts of the network, i.e., links & nodes. In practical cases, the exact value of the unreliability q_k of a component of type k is unknown, $k = 1, 2, \dots, N_k$, where N_k is the total number of available component types, $N_k = \sum_{i=1}^s m_i$. An estimate \tilde{q}_k is usually available on the basis of field failure data, life test data, or empirical models. Unfortunately, these sources of reliability information are subject to uncertainty. Then, the unreliability estimate \hat{q}_k , of which \tilde{q}_k is a realization, is a random variable, and as a result, so is the system reliability estimate $\hat{R}(\hat{\mathbf{q}})$, which is a function of the vector $\hat{\mathbf{q}} = \{\hat{q}_1, \hat{q}_2, \dots, \hat{q}_{N_C}\}$ of the N_C components unreliability estimates \hat{q}_l , $l = 1, 2, \dots, N_C$. Therefore, an informative evaluation of the network reliability

performance entails the computation not only of the expectation, $E[\hat{R}]$, but at least also of the variance, $Var(\hat{R})$, of the reliability estimate, \hat{R} .

In practice, analytical approaches to the network reliability estimation problem can be conducted if some simplifying assumptions are made. For example, the requirement that the components' reliability estimates be statistically independent would allow the analytical estimation or approximation of $Var(\hat{R})$ for some systems. However, the required assumptions, which need to be introduced, are often unrealistic. Concerning the previous example, in practice it is common to use the same type of component in different network locations due to the need for similar performance requirements: the estimated reliability for each component type is then used everywhere that type of component appears in the system design, thus introducing a dependence which is difficult to address analytically. Conversely, the MC approach can readily accommodate the case of repeated components straightforwardly.

To use the MC approach for estimating the reliability of a network, and its associated variance, information on the probability distributions of the unreliability estimates \hat{q}_k , $k = 1, 2, \dots, N_k$, must be available. In this case, we assume to have a realization \tilde{q}_k of the random unreliability estimate \hat{q}_k , and its corresponding estimated variance $V\hat{a}r[\tilde{q}_k]$, for each component type $k = 1, 2, \dots, N_k$. $V\hat{a}r[\tilde{q}_k]$ can be based on empirical data, previous experience, or other sources.

The binomial distribution, $B(n_k | n_t^k, q_k)$, provides the probability of observing n_k failures in n_t^k tests of the type- k components, when the probability of a failure in one test is q_k . The unreliability estimate \hat{q}_k is estimated as n_k/n_t^k . The uncertainty on \hat{q}_k , $k = 1, 2, \dots, N_k$ needs to be propagated to the system unreliability estimate \hat{R} . The two unknown parameters of the underlying Binomial distribution, n_t^k & q_k , can be determined on the basis of the available information \tilde{q}_k & $V\hat{a}r[\tilde{q}_k]$. In practice, the realization \tilde{q}_k can be used as an approximation of the unknown value of the type- k component unreliability, q_k . For the other unknown parameter n_t^k , we consider that, as a consequence of the binomial assumption, the unknown variance $Var[\hat{q}_k]$ of the unreliability estimate, $\hat{q}_k = n_k/n_t^k$, is given by

$$Var[\hat{q}_k] = \frac{1}{(n_t^k)^2} n_t^k q_k (1 - q_k) = \frac{q_k(1 - q_k)}{n_t^k}. \quad (1)$$

An approximate (pseudo) sample size \tilde{n}_t^k for component kind k can now be determined by replacing q_k with the available realization \tilde{q}_k , and n_t^k with \tilde{n}_t^k . By solving (1) with respect to \tilde{n}_t^k , we obtain

$$\tilde{n}_t^k = \frac{\tilde{q}_k(1 - \tilde{q}_k)}{V\hat{a}r[\tilde{q}_k]} \quad (2)$$

where the two available estimates, \tilde{q}_k & $V\hat{a}r[\tilde{q}_k]$, replace the two unknown quantities, q_k & $Var[\hat{q}_k]$.

The MC estimation of the expected value $E[\hat{R}]$, and variance $Var(\hat{R})$ of the system reliability proceeds as follows:

1. Determine the cut-sets of the network. (In practice, this can be difficult for some network systems. There are many references dedicated to cut-set determination. The research in this paper is limited to those networks where the cut-sets can be explicitly obtained.)

2. For each component kind j employed in the network, $j = 1, 2, \dots, N_j \leq N_k$, sample a number of failures n_j^* from the binomial distribution with parameters \tilde{n}_t^j , \tilde{q}_j , and obtain a sample realization $\hat{q}_j^* = n_j^*/\tilde{n}_t^j$ of the type- j component unreliability estimate.
3. Perform a MC simulation of N_s network life histories, sampling the components' failures from the \hat{q}_j^* obtained in Step 2; knowing the system cut sets, accumulate the number of system failures N_f^* , and estimate the system reliability $\hat{R}^* = 1 - (N_f^*/N_s)$.
4. Repeat Step 3 for N_R other sets of realization of components' unreliability estimates \hat{q}_j^* , $j = 1, 2, \dots, N_j \leq N_k$, from Step 2, each time performing a simulation of N_s system lives to estimate the corresponding system reliability $\hat{R}^* = 1 - (N_f^*/N_s)$, which depends on the components' unreliability values \hat{q}_j^* , $j = 1, 2, \dots, N_j$, from which the components' failures are sampled during the simulation.

The N_R values of \hat{R}^* thus obtained can be used to compute estimates of the system expected reliability $E[\hat{R}] = \sum_{r=1}^{N_R} \hat{R}_r^*/N_R$, and associated variance $Var(\hat{R}) = \sum_{r=1}^{N_R} (\hat{R}_r^* - E[\hat{R}])^2 / (N_R - 1)$, which account for the uncertainty in the estimates of the components' unreliabilities \hat{q}_l , $l = 1, 2, \dots, N_C$. Note that, different from analytical approaches, the MC procedure does not encounter difficulties when repeated components appear in the system. In fact, realizations of the unreliability estimates are sampled for each component type, and the same value is used wherever that component type appears in the network design. Alternatively, most analytical estimates require independent reliability estimate assumptions.

IV. GENETIC ALGORITHM OPTIMIZATION APPROACH

In the following sections, we provide some basics of the GA optimization approach with reference to the traditional single-objective GA [9], and then present the generalization of the approach to multi-objective problems [21], [22].

A. Single Objective Genetic Algorithms

GA, first formalized as an optimization method by Holland [8], are search tools modeled after the genetic evolution of natural species. GA differ from most optimization techniques because of their global searching effectuated by one population of solutions rather than from one single solution. They still can guarantee global optimality only asymptotically. Every prospective solution is represented by a vector \mathbf{x} of the independent variables, which is coded in a so-called chromosome (called a genotype), consisting of so-called genes, each one coding one component of \mathbf{x} . A binary coding is widely used.

The GA search starts with the creation of a random initial population of N_{ga} chromosomes, i.e., potential solutions to the problem. Then, these individuals are evaluated in terms of their "fitness" values, i.e., their corresponding objective function values. This initial population is allowed to evolve in successive generations through the following steps:

- 1) selection of a pair of individuals as parents;
- 2) crossover of the parents, with generation of two children;
- 3) genetic mutation;

- 4) replacement in the population, so as to maintain the population number N_{ga} constant.

Every time a new solution \mathbf{x} is proposed by the GA, the objective function is evaluated, and a ranking of the individuals in the current population is dynamically updated, based on their fitness values. This ranking is used in the selection procedure so that, in the long run, the best individuals will have a greater probability of being selected as parents. This is to resemble the natural principles of the "survival of the fittest." Similarly, the ranking is used in the replacement procedures to decide who, among the parents & the children solutions, should survive in the next population. An algorithm based on these procedures is often referred to as a steady-state GA [24].

When using GA, sufficient genetic diversity among solutions in the population should be guaranteed. Lack of such diversity would lead to a reduction of the search space spanned by the GA, and consequently to a degradation of its optimization performance with selection of mediocre individuals resulting in premature convergence to a local minimum. Alternatively, an excess of genetic diversity, especially at later generations, may lead to a degradation of the optimization performance, resulting in very late, or even no, convergence. In the applications which follow in this paper, genetic diversity is maintained by the mutation procedure, but also by resorting to the 'children-parents' replacement procedure according to which the children always replace the parents in the population [9].

B. Multi-Objective Genetic Algorithms

The multi-objective optimization problem arises when, in evaluation of each point \mathbf{x} in the search space, we must consider several objective functions, $f_h(\mathbf{x})$, $h = 1, 2, \dots$, and then identify \mathbf{x}^* , which provides the best compromise among the various objective functions. The comparison of two solutions with respect to several objectives may be achieved through the introduction of the concepts of *Pareto optimality*, and *dominance* [21], [22] which enable solutions to be compared & ranked without imposing any a priori measure as to the relative importance of individual objectives, neither in the form of subjective weights nor arbitrary constraints.

Consider N different objective functions $f_h(\mathbf{x})$, $h = 1, 2, \dots, N$ where \mathbf{x} represents the vector of decision variables identifying a generic solution proposal. We say that solution \mathbf{x} *dominates* solution \mathbf{y} if the following inequalities are valid [25]: $f_h(\mathbf{x}) \geq f_h(\mathbf{y})$ for all $h = 1, 2, \dots, N$, and $f_j(\mathbf{x}) > f_j(\mathbf{y})$ for some j .

The solutions not dominated by any other are said to be *non-dominated solutions*. Within the evolutionary approach, in order to treat simultaneously several objective functions, it is necessary to generalize the single-fitness procedure employed in the single-objective GA by assigning N fitness values to each \mathbf{x} .

Concerning the insertion of an individual (i.e., an \mathbf{x} value) in the population, often constraints exist which impose restrictions that the candidate individual has to satisfy, and whose introduction speeds up the convergence of the algorithm, due to a reduction in the search space. Such constraints may be satisfied, just as in the case of single-objective GA, by testing whether, in the course of the population creation & replacement procedures, the candidate solution fulfills the criteria pertaining to all the N fitness measures.

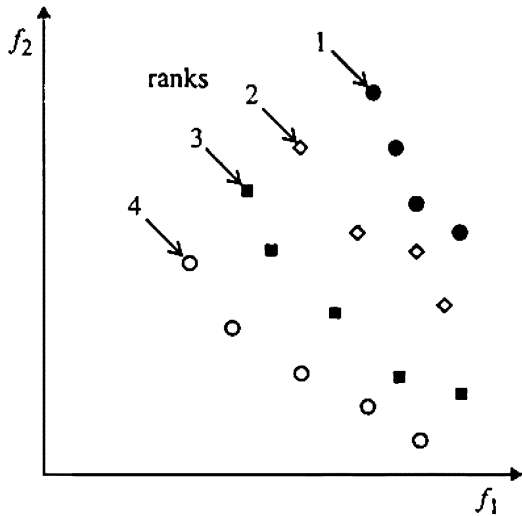


Fig. 1. Example of population ranking for a problem of maximization of f_1 and f_2 .

Once a population of individuals (chromosomes) has been created, we rank them according to the Pareto dominance criterion by considering the N -dimensional fitness space, $f_h(\mathbf{x})$, $i = 1, 2, \dots, N$ (see Fig. 1 for $N = 2$). All nondominated individuals in the current population are identified. These solutions are considered to be the best, and assigned the rank 1. Then, they are temporarily & virtually removed from the population, and the next set of nondominated individuals are identified, and then assigned rank 2. This process continues until every solution in the population has been ranked.

The selection & replacement procedures of the multi-objective genetic algorithms are based on this ranking. Every chromosome belonging to the same rank class has to be considered equivalent to any other of the class, i.e., it has the same probability of the others to be selected as a parent, and survive subsequent replacement. Alternatively, rank-equivalent individuals could be assigned a single, dummy fitness value, which can then be further manipulated to maintain diversity in the population [20].

During the optimization search, we record & update an archive of vectors, each one constituted by a nondominated chromosome, and by the corresponding N fitness measures, representing the dynamic Pareto optimality surface. At the end of each generation, nondominated individuals in the current population are compared, in terms of the fitness measures, with those already stored in the archive, and the following archival rules are implemented:

1. If the new individual dominates existing members of the archive, these are removed, and the new one is added.
2. If the new individual is dominated by any member of the archive, it is not stored.
3. If the new individual neither dominates nor is dominated by any member of the archive, then:
 - if the archive is not full, the new individual is stored; or
 - if the archive is full, the new individual replaces the *most similar* one in the archive.

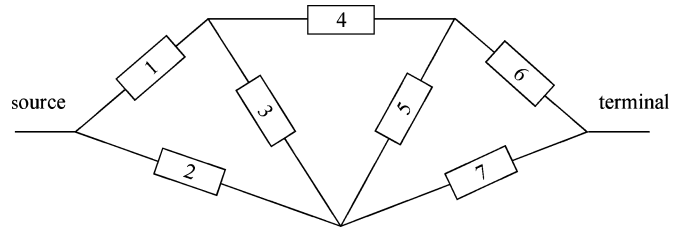


Fig. 2. 7-Component network system.

Similarity criteria are based on an appropriate concept of distance being introduced to measure the similarity between two individuals. In this paper, we shall adopt a Euclidean distance based on the fitness values of the solution chromosomes normalized to the respective mean values in the archive.

The archive of nondominated individuals is also exploited by introducing an elitist parent selection procedure that should, in principle, be more efficient [21]. Every individual in the archive (or, if the archive size is too large, a pre-established fraction of the population size N_{ga} , typically $N_{ga}/4$) is chosen once as a parent in each generation. The logic behind this selection procedure is to guarantee a better propagation of the genetic code of nondominated solutions, and thus a more efficient evolution of the population toward Pareto optimality.

At the end of the search procedure, the result of the optimization is constituted by the archive itself which contains the Pareto optimality region, i.e., the Pareto-optimal set of solutions.

V. EMBEDDING THE MONTE CARLO SIMULATION IN THE GENETIC ALGORITHM SEARCH ENGINE

In practical cases, the search for an optimal network design (redundancies and/or component types) involves a choice among a large number of potential alternatives. Running a full Monte Carlo simulation (several thousands of life histories) for each of them is impractical. If the search for the optimal solution is performed by means of a GA, a Monte Carlo code should be run for each individual of the chromosomes' population considered in the successive generations of the search. This would be very time consuming, and inefficient.

A possible solution to this problem follows from the consideration that, during the GA search, the best chromosomes appear a large number of times in the successive generations, whereas the bad solutions are readily eliminated [12]. Thus, for each proposed chromosome, we can run a limited number of MC trials. More precisely, in our calculations, for each chromosome proposed by the GA, the network expected reliability & variance estimates are obtained on the basis of $N_R = 10$ samples of the N_C component unreliability estimates \hat{q}_l , $l = 1, 2, \dots, N_C$, and for each set, $N_S = 1,000$ MC trials are performed. During the GA evolution, the archive of the best chromosome-solutions obtained in previous MC runs, and the corresponding MC objective function estimates, are updated. Whenever a chromosome is re-proposed, the newly computed objective function estimates can be accumulated with those stored in the archive, and the large number of times a 'good' chromosome is proposed by natural selection allows accumulating repeatedly over the results of the few-histories runs, thus achieving at the end s -significant results [26], [27]. This way of proceeding also avoids wasting

TABLE I
EXAMPLE 1 COMPONENT PARAMETERS

node pair 1 ($i = 1$)						node pair 2 ($i = 2$)					node pair 3 ($i = 3$)				
j	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_i^{ij}	c_{ij}	w_{ij}	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_i^{ij}	c_{ij}	w_{ij}	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_i^{ij}	c_{ij}	w_{ij}
1	0.97	0.0029	10	22	11	0.94	0.0282	2	21	12	0.95	0.0238	2	19	11
2	0.92	0.0184	4	17	9	0.85	0.0159	8	18	6	0.89	0.0163	6	15	9
3	0.81	0.0128	12	11	5	0.79	0.0151	11	10	6	0.81	0.0192	8	9	5
4	0.68	0.0242	9	8	4	0.70	0.0191	11	9	3	0.62	0.0236	10	7	5

node pair 3 ($i = 3$)						node pair 4 ($i = 4$)					node pair 5 ($i = 5$)				
j	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_i^{ij}	c_{ij}	w_{ij}	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_i^{ij}	c_{ij}	w_{ij}	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_i^{ij}	c_{ij}	w_{ij}
1	0.93	0.0136	4	24	14	0.92	0.0082	9	18	12	0.96	0.0326	8	19	13
2	0.90	0.0150	6	19	8	0.88	0.0075	14	14	9	0.92	0.0213	8	17	10
3	0.82	0.0148	10	12	6	0.77	0.0295	6	9	6	0.80	0.0191	14	12	7
4	0.72	0.0202	10	8	5	0.62	0.0157	15	7	5	0.77	0.0163	12	9	6

node pair 7 ($i = 7$)					
j	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_i^{ij}	c_{ij}	w_{ij}
1	0.93	0.0326	2	19	13
2	0.85	0.0213	6	17	10
3	0.78	0.0191	9	12	7
4	0.65	0.0163	14	9	6

NOTE: the reliability estimates $\tilde{r}_{ij} = 1 - \tilde{q}_{ij}$ of alternative j of node pair i are reported in place of the unreliability \tilde{q}_{ij} and σ_{ij}^2 stands for its estimated variance, $Var[\tilde{r}_{ij}]$.

time on ‘bad’ configurations which are simulated only a small number of times. We call this approach ‘drop-by-drop simulation’ for its similarity to this way of filling a glass of water.

Finally, to enhance the s -significance of the estimates, at the end of each generation the objective functions’ estimates of the (nondominated) solutions in the archive are reinforced by sampling $N_R = 50$ additional sets of estimates $\hat{q}_l, l = 1, 2, \dots, N_C$, and running $N_S = 1,000$ system life histories for each set $R = 1, 2, \dots, 50$.

VI. NUMERICAL EXAMPLES

A. Example 1

The first example describes a network design problem with multiple choices of component types available for each link, and the possibility of allocating one additional redundant component in certain links; thus, for each node pair, a maximum redundancy of two components is allowed. This problem is of practical interest because often low level redundancy between node pairs is preferred in network applications. The example network has $s = 7$ links, as depicted in Fig. 2.

Each link i has $m_i = 4$ functionally equivalent alternatives (i.e., component types) which can be used $i = 1, 2, \dots, s$. Their characteristic parameters are listed in Table I. The design requires that at least one component be implemented for each link. The design objective is to maximize the network reliability while minimizing the variance of its estimate. The constraints of the system are a maximum cost of 85, and maximum weight of 58, in arbitrary units.

The multi-objective GA, combined with a MC evaluation of the network reliability estimates, is used to determine the optimal network design. The objective functions are the expecta-

TABLE II
ALTERNATIVE DESIGN CHOICES FOR A SAMPLE NODE PAIR

Node pair i design alternatives	1	2	3	4	1,1	1,2	1,3	1,4	2,2	2,3	2,4	3,3	3,4	4,4
(j) or $(j, k), j, k=1,2,3,4$														
Design label (z)	1	2	3	4	5	6	7	8	9	10	11	12	13	14

NOTE: j indicates that link i is made of a single component of type j ; j, k indicates a redundancy of two components of types j and k , respectively.

TABLE III
GENETIC ALGORITHM PARAMETERS AND RULES, EXAMPLE 1

Number of chromosomes (population size)	100
Number of generations (termination criterion)	100
Selection	Fit-Fit
Replacement	Children-parents
Number of generation without elitist selection	10
Fraction of parents chosen with elitist selection	0.25
Mutation probability	0.001
Crossover probability	1
Non-dominated chromosomes in the final archive	3

tion of the system reliability $E[\hat{R}]$, and the negative of its variance $-Var(\hat{R})$.

The chromosomes are composed of seven genes, each one coding a possible component type $j = 1, 2, 3, 4$, and redundancy level for each node pair $i = 1, 2, \dots, 7$. Table II lists the possible design alternatives for a sample node pair i , under

TABLE IV
SYSTEM RELIABILITIES \hat{R} AND VARIANCES $Var(\hat{R})$ OF THE THREE NONDOMINATED SOLUTIONS

	$E[\hat{R}]$	$\sigma_{\hat{R}}^{MC}$	$Var(\hat{R})$	$\sigma_{Var(\hat{R})}^{MC}$	Selected component(s) type(s) $j (j, k)$ for each node pair $i, i=1,2,\dots,7$						
					1	2	3	4	5	6	7
1	0.9667	6.38E-2	4.915E-3	3.230E-5	3	1	4	4	3	4	1
2	0.9657	1.53E-2	3.287E-3	5.226E-6	3	3,4	3	4	3	4	1
3	0.9618	1.00E-2	1.053E-3	1.840E-6	3	3,4	3	4	3	1	3

NOTE: $\sigma_{\hat{R}}^{MC}$ and $\sigma_{Var(\hat{R})}^{MC}$ denote the standard deviations of the MC estimates of the objective functions; j, k indicates a redundancy of two components made with types j and k .

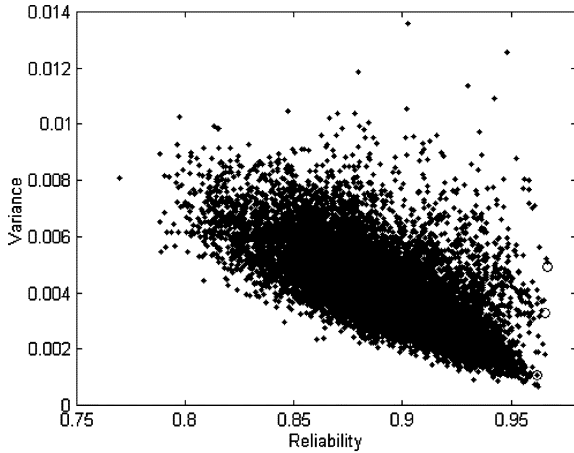


Fig. 3. MAKE \diamond into BLACKdiamond!!!! Example 1 enumerated search space. \bullet = search space for the multiobjective problem; \circ = nondominated solutions found by the GA.

the design assumptions ($m_i = 4$ possible types of components available, and a maximum redundancy of two components allowed for each node pair), $i = 1, 2, \dots, s$. The alternative designs are labeled with an index $z, z = 1, 2, \dots, 14$. Genes with 4 bits are used to represent all 14 possible designs ($2^4 = 16$). If during the GA evolution a nonexistent node pair design is proposed, i.e., design 15 or 16, the corresponding chromosome is discarded. The particular GA rules & control parameters, used for Example 1, are reported in Table III.

The GA was applied to this example. Three Pareto optimal solutions were identified. Table IV provides details describing the Pareto optimal set of solutions. Design Solution 1 is characterized by the highest value of the expected estimated reliability, but also the highest uncertainty in such estimate. Design Solution 2 differs from Design Solution 1 only for the choices made with reference to the second & third links. In particular, the second link is more critical for the network than the third one because the former provides, in connection with the seventh link, the shortest path from source to terminal. In the Design Solution 2, the expected reliability of the redundancy (3, 4) in the second link turns out to be equal to 0.9370, which is slightly lower than that of the type-1 component alone, 0.94, used in Design Solution 1; instead, the variance is an order of magnitude lower (i.e., 0.0025 against 0.0282). On the contrary, at the less crucial third link, the type-3 component chosen in Design Solution 2 has a higher reliability (0.82), and lower variance (0.0148) than that of type 4 in Design Solution 1 (0.72 and 0.0202, respectively). The combination of these effects leads to

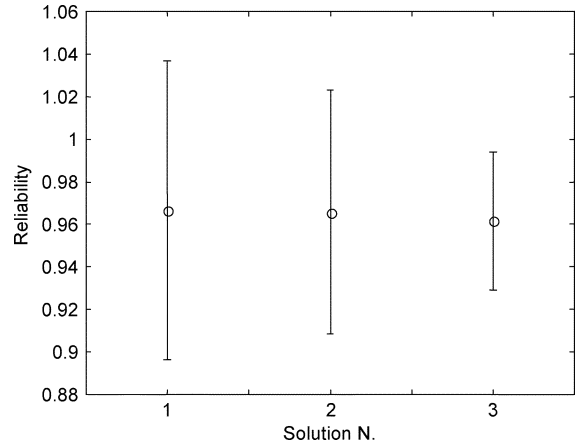


Fig. 4. Expected network reliability estimate $E[\hat{R}]$, with corresponding error bar of one standard deviation for the three Pareto optimal solutions.

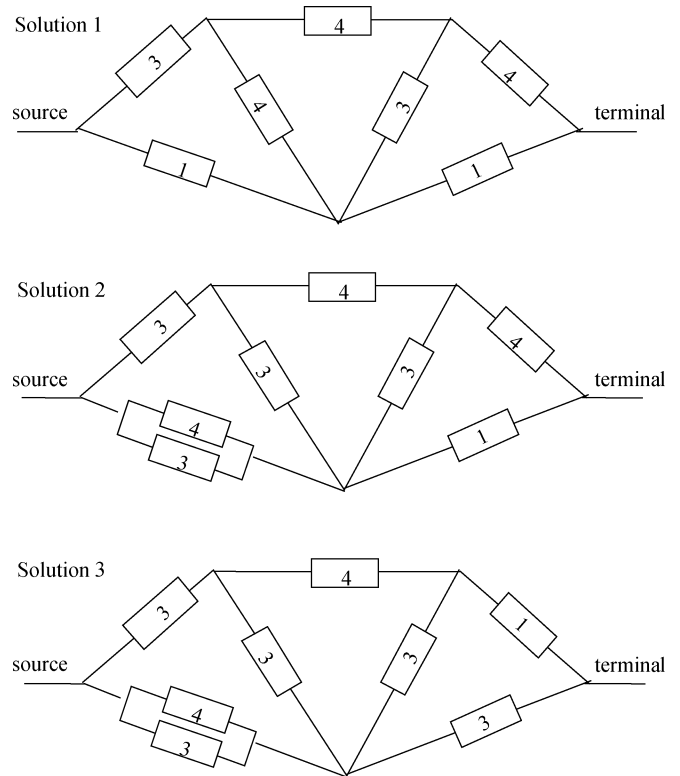


Fig. 5. Example 1 Pareto optimal solution set.

a lower expected overall network reliability, but lower variance, of Design Solution 2.

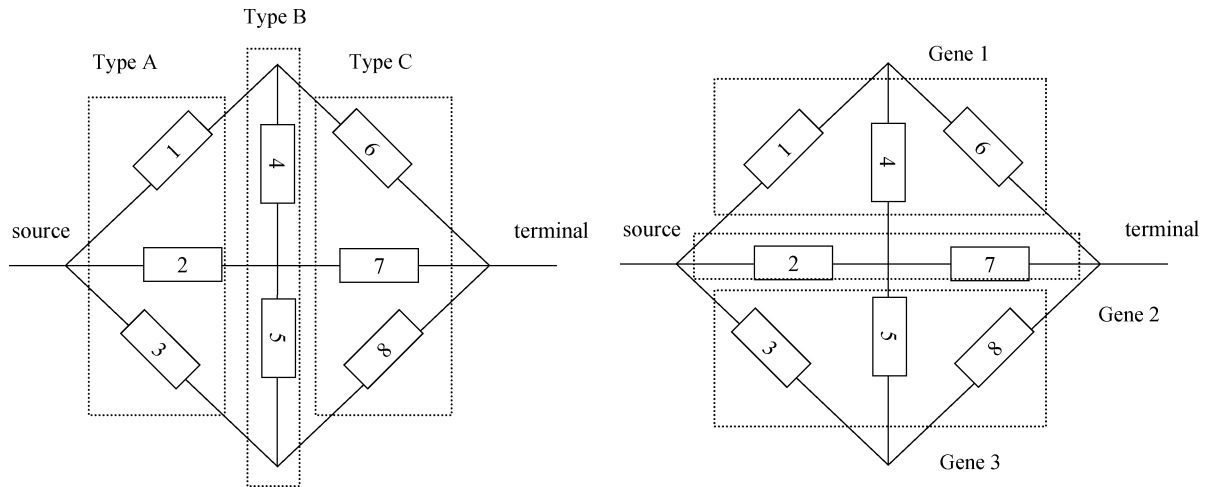


Fig. 6. 8-Component network system: grouping of components by type (left), and genes (right).

Comparing Design Solutions 2 & 3, the choice in the latter of the more reliable, but uncertain, type-1 component in the sixth link, and of the less reliable, but less uncertain, type-3 component in the seventh link produces an overall slightly lower network reliability estimate for design 3, but with a significantly lower estimation uncertainty.

For validation purposes, the complete search space of approximately 10^8 alternatives was spanned, and the MC estimates of the expectation of the system reliability $E[\hat{R}]$, and of its variance $Var(\hat{R})$, were computed for the 15 833 solutions satisfying the constraints on cost & weight (Fig. 3). This required 600 minutes of CPU time on an ATHLON 1000 MHz computer. Because we were eventually aiming at validating the Pareto front found by the GA at the end of the multi-objective search, and given the simplicity of the calculations, it was simpler, although somewhat costly from the computational point of view, to evaluate by exhaustive enumeration all the feasible alternatives, rather than implement other known multi-objective optimization techniques to compare their corresponding results to those found by the GA. Fig. 3 also shows the three nondominated solutions ('circles'), identified by the genetic algorithm, which are confirmed to lie on the frontier of Pareto optimal performance (the lower right region of the Figure, characterized by solutions with high reliability, and low variance). The CPU time required for the GA search was 14 minutes on the same computer. The difference in CPU times required by the enumerative search, and the GA/MC approach, underlines the efficiency of the latter in quickly driving the search toward the portion of interest of the search space.

Fig. 4 provides additional details describing the Pareto optimal set of solutions. Fig. 4 presents the MC estimated values of $E[\hat{R}]$, with error bars of one standard deviation, for the three nondominated solutions contained in the archive at convergence. In the figure, they are ordered in decreasing values of $E[\hat{R}]$. The corresponding actual values are listed in Table IV, together with the MC estimates of $Var[\hat{R}]$, and the corresponding network designs. Fig. 5 presents pictorially the three optimal design alternatives.

Given the nondominated design solutions, characterized by different levels of network reliability & uncertainty, the

decision makers can then select, according to their respective tolerance for risk, a preferred alternative with respect to the two objective functions. If the decision relied only on the expected value of the system reliability, the best solution would be number 1 in Table IV. In reality, decision makers are often risk-averse, in the sense that they are not completely satisfied with expected predictions, but rather they want the highest possible confidence that the actual performance is the one predicted. As an outcome of the multi-objective approach, the decision-maker choice could be shifted toward solution number 3, which presents lower uncertainty on an only slightly worse system reliability estimate.

B. Example 2

The network in Fig. 6 has $s = 8$ node pairs & corresponding links. To link each node pair $i = 1, 2, \dots, 8$, there are $m_i = 4$ alternative component-types $j = 1, 2, \dots, 4$ which are functionally equivalent, but with different reliability, cost, and weight characteristics (Table V). Node Pairs 1, 2, and 3 share the same kind of component, i.e., Group A; for Node Pairs 4 and 5, the components can be chosen within Group B; Node Pairs 6, 7, and 8 are selected from Group C. As before, the objective is to determine, for each node pair, the principal component type, and one potential redundant component (if any), such that the system reliability is maximized, while the variance of the system reliability estimate is minimized. The maximum allowable network cost is 50, and the maximum allowable weight is 50.

The system is physically decomposed in three subsystems constituted by Node Pairs (1, 4, 6), (2, 7), and (3,5,8), respectively. In the GA methodology, each proposed solution of the system configuration is coded into bit-string chromosomes; three genes are used to code the configurations proposed for the three subsystems into which the network has been decomposed (Fig. 6, right). Furthermore, when generating two offspring from the selected parents, we wish to maintain a substantial part of the parents' genetic content in the offspring by changing only three node pairs, one for each subsystem, out of the eight constituting the whole system. Sub-genes are then used to code the proposed choice for each node pair, with the same

TABLE V
EXAMPLE 2 COMPONENT PARAMETERS

j	Component Group A Node Pairs ($i = 1, 2, 3$)					Component Group B Node Pairs ($i = 4, 5$)					Component Group C Node Pairs ($i = 6, 7, 8$)				
	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_t^{ij}	c_{ij}	w_{ij}	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_t^{ij}	c_{ij}	w_{ij}	\tilde{r}_{ij}	σ_{ij}^2	\tilde{n}_t^{ij}	c_{ij}	w_{ij}
1	0.93	0.0251	3	12	12	0.88	0.0282	4	12	10	0.80	0.0838	2	6	7
2	0.72	0.0153	13	10	10	0.71	0.0132	16	8	8	0.71	0.0263	8	5	5
3	0.65	0.0091	25	6	6	0.67	0.0094	24	5	4	0.66	0.0114	20	3	4
4	0.62	0.0057	41	5	5	0.66	0.0080	28	4	4	0.60	0.0052	46	4	3

NOTE: the reliability estimates $\tilde{r}_{ij} = 1 - \tilde{q}_{ij}$ of alternative j of node pair i are reported in place of the unreliability \tilde{q}_{ij} and σ_{ij}^2 stands for its estimated variance, $Var[\tilde{r}_{ij}]$.

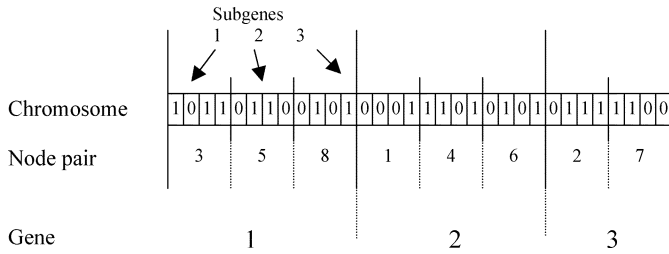


Fig. 7. Chromosome employed for coding the network design.

TABLE VI
GENETIC ALGORITHM DATA AND RULES, EXAMPLE 2

Number of chromosomes (population size)	100
Number of generations (termination criterion)	100
Selection	Fit-Fit
Replacement	Children-parents 50% Fittest 50%
Number of generation without elitist selection	10
Fraction of parents chosen with elitist selection	0.25
Mutation probability	0.001
Crossover probability	1
Non-dominated chromosomes in the final archive	8

coding employed in Example 1. Fig. 7 shows an example of a resulting chromosome; to generate the children in the crossover operation, the entire genes are swapped between the parents. The relevant GA data used for Example 2 are given in Table VI.

The search space of 5.67×10^{13} alternatives was efficiently spanned by the GA search in 20 minutes of CPU time on an ATHLON 1000 MHz computer. Enumeration of all alternatives was not practical, and was not performed.

Figs. 8 & 9 present the results of the search. In Fig. 8, we report the values of the expected system reliability $E[\hat{R}]$, and of its variance $Var(\hat{R})$, for the nondominated solutions contained in the archive at convergence. In Fig. 9, we report the corresponding values of $E[\hat{R}]$, and of the standard deviations ordered in decreasing values of $E[\hat{R}]$. Both Figures, and the actual values reported in Table VII, reveal that the Pareto optimal solutions can be grouped into two distinctly separate clusters that differ

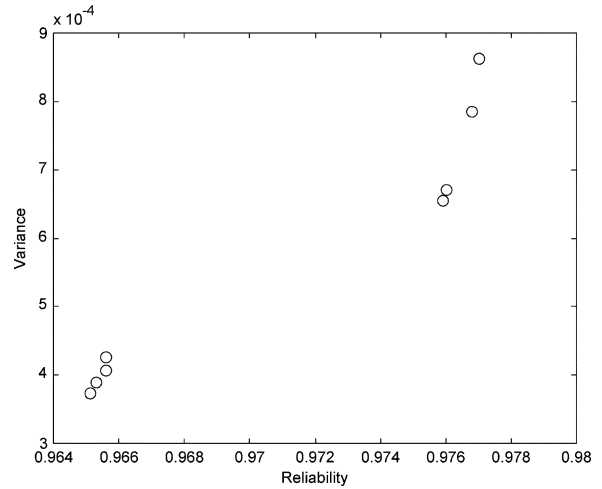


Fig. 8. Multiobjective optimization results, Example 2.

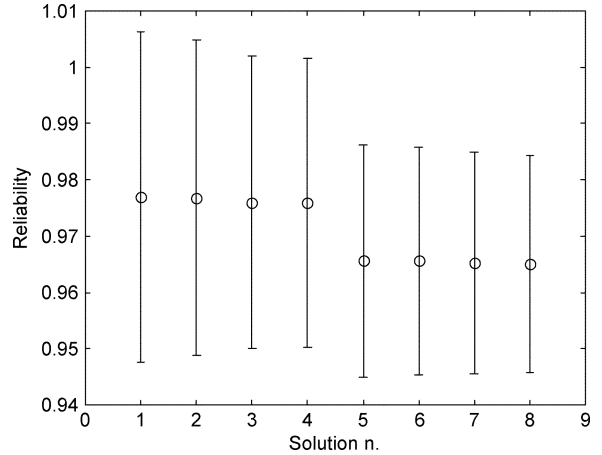


Fig. 9. Expected network reliability estimate $E[\hat{R}]$, with corresponding error bar of one standard deviation for the eight Pareto optimal solutions.

for the design choice in correspondence of link $i = 2$. When the first & most reliable alternative for this link is selected ($i = 2$, $j = 1$ in Table V), higher values of $E[\hat{R}]$ are attained (Solutions 1–4 in Table VII), but the larger uncertainty on its reliability estimate, due to the small sample size available ($\tilde{n}_t^{2,1} = 3$), propagates to a larger uncertainty on the system reliability. Alternatively, Solutions 5–8 in Table VII, characterized by the redundancy (3,4) in link $i = 2$ with lower but more precise unreliability estimates, present slightly lower but less uncertain values

TABLE VII
EXPECTED VALUES OF NETWORK RELIABILITY AND VARIANCES FOR NONDOMINATED SOLUTIONS

Solution	$E[\hat{R}]$	$\sigma_{\hat{R}}^{MC}$	$Var(\hat{R})$	$\sigma_{Var(\hat{R})}^{MC}$	Selected component for each node pair							
					1	2	3	4	5	6	7	8
1	0.9770	5.904E-03	8.622E-04	4.608E-06	3	1	4	4	4	4,4	2,3	3
2	0.9768	4.643E-03	7.858E-04	$< 10^{-7}$	4	1	3	4	4	4	2,4	3,3
3	0.9760	2.351E-03	6.707E-04	$< 10^{-7}$	3	1	4	4	4	3,4	2	3,4
4	0.9759	7.776E-03	6.561E-04	5.103E-06	4	1	3	4	4	3,4	2	3,4
5	0.9656	3.014E-02	4.274E-04	$< 10^{-7}$	3	3,4	4	3	3	3,4	2,3	3
6	0.9656	1.837E-03	4.082E-04	$< 10^{-7}$	3	3,4	3	4	3	3,4	2,3	3
7	0.9653	1.949E-03	3.901E-04	$< 10^{-7}$	3	3,4	4	4	3	3,2	3,4	2
8	0.9651	5.728E-03	3.734E-04	2.139E-06	3	4,4	3	4	4	3	2,4	3,4

NOTE: $\sigma_{\hat{R}}^{MC}$ and $\sigma_{Var(\hat{R})}^{MC}$ denote the standard deviations of the MC estimates of the objective functions; j, k indicates a redundancy of two components made with types j and k .

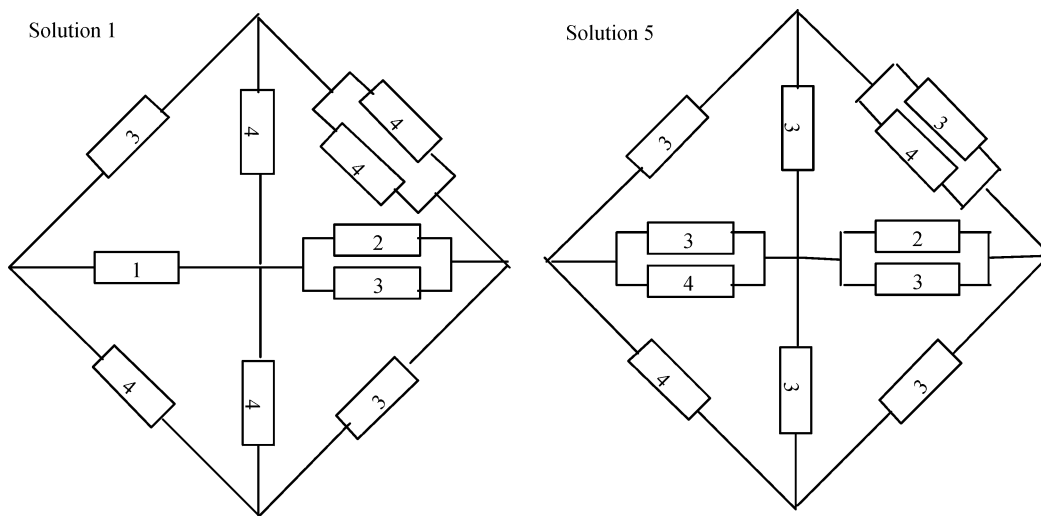


Fig. 10. Pareto optimal network designs belonging to the two clusters: design 1 (left), and design 5 (right).

of network unreliability. Fig. 10 shows two sample Pareto optimal network designs belonging to the two clusters: Design 1 (left), and Design 5 (right).

Overall, the results of the multi-objective approach lead to considerations analogous to those observed in the first example. On the basis of a single objective optimization, the network optimal design would be Solution 1. When the decision-maker is provided with the more informative multi-objective solution information, he/she can take his/her decision accounting for the uncertainties in the system performance. Indeed, a risk-averse decision should concentrate on solutions belonging to the second cluster, which present lower uncertainties on the system performance.

VII. CONCLUSIONS

Due to the growing exploitation of highly dimensioned & complex network systems, (telecommunication, and others), the tasks of designing & operating highly reliable networks are acquiring increasing significance.

Estimation of the reliability of a network system is usually based on estimates of the reliability parameters of its constituent lower-level components (nodes & links). In practice, these latter

reliability estimates are obtained on the basis of limited data so that they are affected by significant uncertainty. Inevitably, these uncertainties propagate to the network reliability estimate. Then, if we wish to optimize the reliability performance of a network system, the mere maximization of its reliability estimate, regardless of the associated uncertainty, may not be satisfactory. Indeed, often in practice the decision makers are risk-averse, in the sense that they usually would prefer designs possibly sub-optimal with respect to the system reliability, if lower estimation uncertainty is assured.

In this paper, the problem of determining the optimal network design has been addressed within a multi-objective search based on an effective coupling of genetic algorithms, and Monte Carlo simulation, which allows for the explicit consideration of the uncertainties in the network reliability estimates arising from the uncertainties in the components' reliability parameters. The multi-objective point-of-view has been adopted to determine solutions which give optimal reliability with a high degree of confidence, under the existing uncertainties. The GA has been shown to provide an efficient search tool when the number of possible alternatives is very large. As a complement, the Monte Carlo simulation provides a flexible tool for evaluating the two objective functions to be maximized by the genetic algorithm

search: the expectation of the network reliability estimate, and the negative of its variance.

The approach has been applied to two network design problems with multiple choices of components' types available, and the possibility of allocating redundancy. Possible design constraints on total cost & weight are considered. The results achieved show that when the decision-maker is provided with the multi-objective solution information, different solution alternatives may be considered which allow the identification of a risk-averse network design characterized by a high degree of confidence in the actual network reliability.

REFERENCES

- [1] S. Kiu and D. F. McAllister, "Reliability optimization of computer-communication networks," *IEEE Trans. Reliab.*, vol. 37, no. 5, pp. 475–483, Dec. 1988.
- [2] F. Altıparmak, B. Dengiz, and A. Smith, "Reliability optimization of computer communication network using genetic algorithms," in *Proc. IEEE Int. Conf. Systems, Man and Cybernetics*, vol. 5, Piscataway, NJ, USA, 1998, 98CB36218, pp. 4676–4680.
- [3] A. Dhingra, "Optimal apportionment of reliability & redundancy in series systems under multiple objectives," *IEEE Trans. Reliab.*, vol. 41, no. 4, pp. 576–582, 1992.
- [4] K. Misra and U. Sharma, "An effective approach for multiple criteria redundancy optimization problems," *Microelectron. Reliab.*, vol. 31, no. 2/3, pp. 303–321, 1991.
- [5] —, "Multi-criteria optimization for combined reliability and redundancy allocation in systems employing mixed redundancies," *Microelectron. Reliab.*, vol. 31, no. 2/3, pp. 323–335, 1991.
- [6] P. Giuggioli Busacca, M. Marseguerra, and E. Zio, "Multiobjective optimization by genetic algorithms: application to safety systems," *Reliab. Eng. Syst. Saf.*, vol. 72, pp. 59–74, 2001.
- [7] R. Rubinstein, G. Levitin, A. Lisniaski, and H. Ben-Haim, "Redundancy optimization of static series-parallel reliability models under uncertainty," *IEEE Trans. Reliab.*, vol. 46, no. 4, pp. 503–511, 1997.
- [8] J. H. Holland, *Adaptation in Natural and Artificial System*. Ann Arbor, MI: University of Michigan Press, 1975.
- [9] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*: Addison-Wesley Publishing Company, 1989.
- [10] D. Beasley, D. Bull, and R. Martin, "An overview on genetic algorithms: part 1. Fundamentals," *University Computing*, vol. 15, no. 2, pp. 58–69, 1993.
- [11] —, "An overview on genetic algorithms: part 2. Research topics," *University Computing*, vol. 15, no. 4, pp. 170–181, 1993.
- [12] D. Fogel, "An introduction to simulated evolutionary optimization," *IEEE Trans. Neural Netw.*, vol. 15, no. 1, pp. 3–14, 1994.
- [13] F. Herrera and J. L. Verdegay, Eds., *Genetic Algorithm and Soft Computing*. Heidelberg: Ohysica-Verlag, 1996.
- [14] P. A. Joyce, T. A. Withers, and P. J. Hickling, "Application of genetic algorithms to optimum offshore plant design," in *Proc. ESREL 98*, Trondheim, Norway, Jun. 16–19, 1998, pp. 665–671.
- [15] D. W. Coit and A. E. Smith, "Use of genetic algorithm to optimize a combinatorial reliability design problem," in *Proc. 3rd IIE Research Conf.*, 1994, pp. 467–472.
- [16] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999.
- [17] E. Zitzler, "Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications," Swiss Federal Institute of Technology (ETH) Zurich, Dec. 1999, TIK-Schriftenreihe Nr. 30, Diss ETH no. 13398, Shaker Verlag, Germany, ISBN 3-8265-6831-1.
- [18] C. M. Fonseca and P. J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization," *Evol. Comput.*, vol. 3, no. 1, pp. 1–16, Spring 1995.
- [19] —, "Genetic algorithms for multiobjective optimization: formulation, discussion and generalization," in *Genetic Algorithms: Proc. 5th Int. Conf.*, S. Forrest, Ed. San Mateo, CA: Morgan Kaufmann, 1993.
- [20] N. Srinivas and K. Deb, "Multiobjective function optimization using nondominated sorting genetic algorithms," *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, 1995.
- [21] G. T. Parks, "Multiobjective pressurized water reactor reload core design using genetic algorithm search," *Nucl. Sci. Eng.*, vol. 124, pp. 178–187, 1997.
- [22] V. G. Toshinsky, H. Sekimoto, and G. I. Toshinsky, "A method to improve multiobjective genetic algorithm optimization of a self-fuel-providing LMFBR by niche induction among nondominated solutions," *Ann. Nucl. Energy*, vol. 27, pp. 397–410, 2000.
- [23] H. Wang and H. Pham, "Survey of reliability and availability evaluation of complex networks using Monte Carlo techniques," *Microelectron. Reliab.*, vol. 37, no. 2, pp. 187–209, 1997.
- [24] S. Martorell, S. Carlos, A. Sanchez, and V. Serradell, "Constrained optimization of test intervals using a steady-state genetic algorithm," *Reliab. Eng. Syst. Saf.*, vol. 67, pp. 215–232, 2000.
- [25] Y. Sawaragi, H. Nakayama, and T. Tanino, *Theory of Multiobjective Optimization*. Orlando, Florida: Academic Press, 1985.
- [26] M. Marseguerra and E. Zio, "Optimizing maintenance and repair policies via a combination of genetic algorithms and Monte Carlo simulation," *Reliab. Eng. Syst. Saf.*, vol. 68, pp. 69–83, 2000.
- [27] M. Marseguerra, E. Zio, and L. Podofilini, "A multiobjective genetic algorithm approach to the optimization of the technical specifications of a nuclear safety system," *Reliab. Eng. Syst. Saf.*, vol. 84, pp. 87–99, 2004.

Marzio Marseguerra (B.S. in engineering, 1954; specialization in nuclear physics, 1956; specialization in applied nuclear physics, in 1957, all from University of Rome) is full professor at the Department of Nuclear Engineering of the Polytechnic of Milan, Italy where he is the head of the laboratory of risk analysis and signal analysis. His current interests include: development of innovative techniques such as neural computing, fuzzy logic, genetic algorithms, biased Monte Carlo simulation for application in reliability and safety analysis, process diagnostics and control, contaminant transport in porous media, safeguard analysis. He is co-author of one international book and of more than 200 papers on international journals and conferences.

Enrico Zio (B.S. in nuclear engineering., Politecnico di Milano, 1991; M.Sc. in mechanical engineering., UCLA, 1995; Ph.D. in nuclear engineering., Politecnico di Milano, 1995; Ph.D. in nuclear engineering., MIT, 1998) is a full professor of Nuclear Engineering at the Politecnico di Milano, Vice-Chairman of the European Safety and Reliability Association, ESRA (since 2000), Chairman of the Italian Chapter of the IEEE Reliability Society. His research topics are: reliability and safety analysis of complex systems under stationary and dynamic operation, by Monte Carlo simulation; Soft computing techniques (neural networks, fuzzy logic, genetic algorithms) for system monitoring, fault diagnosis and optimal design; Stochastic models of radioactive and toxic contaminant transport in groundwater. He is co-author of one international book and of more than 80 papers on international journals.

Luca Podofilini received the Ph.D. degree from the department of Nuclear Engineering at the Polytechnic of Milan. His research activity concerns multi-criteria reliability optimization, reliability prediction with uncertainty, Genetic Algorithms for multi-criteria optimization of system design, Monte Carlo simulation for modeling the dynamics of complex industrial systems, importance measures for multi-state systems. He has been recently granted a Marie Curie Fellowship from the European Community at the Norwegian Institute for Technology, Trondheim, Norway.

David W. Coit is an Associate Professor of Industrial & Systems Engineering at Rutgers University. He received the B.S. degree in mechanical engineering from Cornell University, an MBA from Rensselaer Polytechnic Institute, and the M.S. and Ph.D. degrees in industrial engineering from the University of Pittsburgh. He has over ten years of experience working for IIT Research Institute (IITRI), Rome NY where he was a reliability analyst and project manager, and in his final position, the Manager of Engineering at IITRI's Assurance Technology Center. In 1999, he was awarded a CAREER grant from NSF to study reliability optimization. His current research involves reliability prediction and optimization, and multi-criteria optimization considering uncertainty. He is a member of the IEEE and IEEE Reliability Society.