Computers and Structures 187 (2017) 77–87

Contents lists available at ScienceDirect

Computers and Structures

journal homepage: www.elsevier.com/locate/compstruc

A rank-based constraint handling technique for engineering design optimization problems solved by genetic algorithms

Rafael de Paula Garcia [b], Beatriz Souza Leite Pires de Lima [b,*], Afonso Celso de Castro Lemonge [c], Breno Pinheiro Jacob [a,b]

[a] LAMCSO – Laboratory of Computer Methods and Offshore Systems[1], Avenida Pedro Calmon, S/N, Cidade Universitária, Ilha do Fundão, Caixa Postal 68.506, 21941-596 Rio de Janeiro, RJ, Brazil

[b] PEC/COPPE/UFRJ – Civil Engineering Program, Post–Graduate Institute of the Federal University of Rio de Janeiro, Avenida Pedro Calmon, S/N, Cidade Universitária, Ilha do Fundão, Caixa Postal 68.506, 21941-596 Rio de Janeiro, RJ, Brazil[2]

[c] UFJF – Federal University of Juiz de Fora, Applied and Computational Mechanics Dept., Rua José Lourenço Kelmer, S/n, 36036-330 Juiz de Fora, MG, Brazil[3]

ARTICLE INFO

Article history:
Received 26 September 2016
Accepted 31 March 2017

Keywords:
Optimization
Genetic algorithms
Constraint-handling techniques

ABSTRACT

This work presents a constraint handling technique (CHT) for the solution of real-world engineering optimization problems by evolutionary algorithms. Referred to as the Multiple Constraint Ranking (MCR), it extends the rank-based approach from many CHTs, by building multiple separate queues based on the values of the objective function and the violation of each constraint. This way, it overcomes difficulties found by other techniques when faced with complex problems characterized by several constraints with different orders of magnitude and/or different units.

The MCR follows an "uncoupled" approach where the CHT is not embedded into the optimization algorithm. Extensive studies are performed to assess its accuracy and robustness, compared to six other up-to-date CHTs, all implemented into the same canonical Genetic Algorithm to allow a neutral and unbiased evaluation. The numerical experiments comprise benchmark functions from the IEEE-CEC competitions on constrained optimization, and also classical structural engineering problems. The performance of the CHTs is compared using efficiency measures in terms of nonparametric statistical tests. The results indicate that the MCR is remarkably more accurate and robust for the subset of problems presenting different-magnitude constraints, while remaining very competitive and one of the top-performers for all other benchmark problems comprising the case studies.

1. Introduction

The use of Evolutionary Algorithms (EAs) and other heuristic methods has been quite common in industry, comprising an efficient alternative for the solution of several types of engineering optimization problems [1–9]. One of the most widely acknowledged EA is the well-known Genetic Algorithm (GA) [10,11]; other methods have also been proposed, such as the Particle Swarm Optimization (PSO) [12,13], Artificial Immune Systems (AIS) [14,15], Ant Colony Optimization (ACO) [16], Crow Search algorithm [17], and many others.

Although originally designed to deal with unconstrained search spaces [18,19], EAs have been successfully complemented by constraint-handling techniques (CHTs) to solve constrained problems, guiding the search process to feasible regions and ideally providing solutions that do not violate any constraint [20–22]. One important line of research consists in studying rank-based CHTs. In this context, Runarsson and Yao [23] proposed the Stochastic Ranking (SR) technique that balances objective and penalty functions by a parameter P_f, producing a ranking by comparing adjacent individuals. Later the same authors proposed the Global Competitive Ranking (GCR) [24] which still balances objective and violation function with the parameter P_f, and defines a fitness function depending of two rankings: according to objective function and the sum of violations. Ho and Shimizu [25] proposed a ranking scheme where the individuals are sorted using a function defined according to three rankings, respectively the objective function; the constraint violation values, and the number of violated constraints. More recently the Balanced Ranking Method

* Corresponding author.
E-mail addresses: garcia816@gmail.com (R.P. Garcia), bia@coc.ufrj.br (Beatriz Souza Leite Pires de Lima), afonso.lemonge@ufjf.br (Afonso Celso de Castro Lemonge), breno@lamcso.coppe.ufrj.br (B.P. Jacob).

[1] http://www.lamcso.coppe.ufrj.br.
[2] http://www.coc.ufrj.br.
[3] http://www.ufjf.br/mac/.

http://dx.doi.org/10.1016/j.compstruc.2017.03.023
0045-7949/

(BRM) [26] was proposed where a merged row is built from two rankings: one for feasible and another for infeasible solutions.

All those methods have undoubtedly presented many advantages over earlier CHTs such as the standard static penalization method, aiming to bypass their inherent shortcomings [20] when applied to real-world, complex engineering problems. However, for such problems there are still some issues and challenges related to the definition of the fitness of a given candidate solution. Currently, the main issue might be how to combine into one term all values involved in the evaluation and comparison of the individuals from a given population, i.e. the objective and violation functions that may have different orders of magnitude and/or different units. This would not be an issue for the most usual cases where the ranges of minimum and maximum violation values are known a priori; these cases could be easily dealt with by usual normalization procedures.

However, there are many real-world engineering applications for which such information is not available and cannot be estimated in advance, including for instance practical applications related to offshore structures such as the optimization of risers connected to floating platforms for offshore oil production [9,27,28], or the optimization of subsea pipeline routes [29,30]. For instance, in this latter application the objective function may be defined in terms of the pipeline length; also, in a multi-objective approach another goal would be to maximize oil flow, production and revenue. Moreover, many disparate constraints are specified, such as declivity (measured in degrees); minimum radius of curvature (in meters); structural constraints such as on-bottom stability [31] and fatigue due to vortex-induced vibrations (VIV) in free spans [32]. Constraints may be defined even as non-dimensional quantities, such as the number of identified interferences of the route with seabed obstacles (subsea equipment, flowlines, other pre-existent pipelines, regions with corals or geotechnical hazards). For such problems, existing CHTs would tend to prioritize solutions-individuals that do not violate constraints with higher magnitudes; eventually, even with the use of more advanced CHTs (such as the GCR or BRM) the influence of constraints with lower magnitude may become insignificant, since those techniques merge all violations into a single sum.

It might be argued that the range of violation values can be estimated by inspecting the search space and evaluating candidate solutions prior, or during, the evolutionary optimization process. However, this might involve some drawbacks. Poor estimations might lead to loss of information, compromising the adequate representation of the constraints; many solutions might be located near or beyond the extremes of the estimated range. Additional evaluations might be required to produce reasonable estimations; this might considerably increase the computational costs (due to the complexity of the analysis methods required to assess the structural constraints). Also, this procedure might also require user intervention, which would not be much convenient. Ultimately, for more accurate estimations the optimization process should be run again.

In this context, this work describes a new ranking-based constrained handling technique, referred here as the Multiple Constraint Ranking (MCR). This method is specifically devised to handle constraints with different orders of magnitude and/or different units, without additional computational overhead associated to the estimation of the range of violation values, and without user intervention. To obtain another desirable characteristic of a CHT, i.e. versatility, the MCR follows a so-called "uncoupled" approach where the CHT is not embedded into the optimization algorithm. This allows its implementation along with different evolutionary algorithms, such as in [33] where an ensemble of four well-known uncoupled CHTs was proposed, each with its own population.

Extensive studies on the MCR are presented, by applying it to several benchmark functions (including those from the IEEE competitions on real parameter constrained optimization, and also classical structural engineering problems), and comparing its performance with other up-to-date CHTs: the Adaptive Penalty Method (APM) [34,35], Tournament Selection Method (TSM) [36], Stochastic Ranking (SR) [23], Global Competitive Ranking (GCR) [24], Ho and Shimizu ranking (HSR) [25] and Balanced Ranking method (BRM) [26]. All CHTs were implemented into the same evolutionary algorithm, thus providing a unique environment that allows a fair, neutral and unbiased evaluation of the efficiency of each CHT, and to compare their efficiency without being influenced by the performance of the optimization algorithm. The comparisons are made using efficiency measures in terms of nonparametric statistical tests.

This paper is organized as follows: Initially, Section 2 summarizes the main concepts related to constrained optimization with evolutionary algorithms, and presents a brief description of the compared CHTs. Section 3 presents the MCR and illustrates its main characteristics by a simple example. The full sets of numerical experiments are described in Section 4, followed by an overall assessment of the results in Section 5, while Section 6 presents a critical analysis for each specific set of experiments. Final remarks and conclusions are presented in Section 7.

2. Constrained optimization with evolutionary algorithms

A general constrained optimization problem may be formally defined by considering a r-dimensional search space comprised by a vector of design variables $\mathbf{x} = (x_1, x_2, x_3, \ldots, x_r)$, with components x_i presenting lower and upper bounds $[l_k, u_k]$. The goal is to minimize an objective function $f(\mathbf{x})$, considering inequality and equality constraints (respectively $g_j(\mathbf{x}) \leq 0$ and $h_j(\mathbf{x}) = 0$) that define the feasible region:

$$\begin{aligned} &\text{minimize} && f(\mathbf{x}) \\ &\text{subject to} && g_i(\mathbf{x}) \leqslant 0,\ i = 1,\ldots,m \\ &&& h_j(\mathbf{x}) = 0,\ j = 1,\ldots,p \\ &&& l_k \leqslant x_k \leqslant u_k,\ k = 1,\ldots,q \end{aligned} \tag{1}$$

Repair methods have been devised to keep only feasible candidate solutions (FCS) along the evolutionary process, using domain knowledge to move an infeasible offspring into the feasible set [37,38]. One of the most popular approaches to treat constraints has been to transform a constrained optimization problem into an unconstrained one, by adding penalty functions $p(\mathbf{x})$ to the objective function $f(\mathbf{x})$ whenever any given constraint is violated, thus leading to an "expanded" objective function $F(\mathbf{x})$, usually referred simply as the "fitness" function:

$$F(\mathbf{x}) = f(\mathbf{x}) + p(\mathbf{x}) \tag{2}$$

The simplest penalty function $p(\mathbf{x})$ is the so-called "death-penalty" [20] that assigns arbitrarily large penalty values, or simply discards the infeasible candidate solutions (ICS) from the optimization process. However, this would prevent the search from using valuable information from the infeasible solutions; thus, several CHTs have been devised to maintain and manage the ICS that unavoidably arise along the search process. The classical *static penalty* technique consists of representing the penalty term $p(\mathbf{x})$ as the sum of values for violation functions $v_j(\mathbf{x})$ associated to each constraint, proportional to the degree of violation, and affected by positive constants – the *penalty factors* k_j that scale and/or weight the relative importance, or degree of severity, of the constraints. Considering for instance the m inequality constraints g, we have: