View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Universidade do Minho: RepositoriUM

Multistart Hooke and Jeeves Filter Method for Mixed Variable Optimization

Florbela P. Fernandes[*,†], M. Fernanda P. Costa[**,†], Edite M.G.P. Fernandes[‡] and Ana Maria A.C. Rocha[‡]

[*] *Department of Mathematics, Polytechnic Institute of Bragança, Portugal*
[†] *CMAT-University of Minho*
[**] *Department of Mathematics and Applications, University of Minho, Portugal*
[‡] *Algorithm R&D Centre, University of Minho, Portugal*

Abstract. In this study, we propose an extended version of the Hooke and Jeeves algorithm that uses a simple heuristic to handle integer and/or binary variables and a filter set methodology to handle constraints. This proposal is integrated into a multistart method as a local solver and it is repeatedly called in order to compute different optimal solutions. Then, the best of all stored optimal solutions is selected as the global optimum. The performance of the new method is tested on benchmark problems. Its effectiveness is emphasized by a comparison with other well-known stochastic solvers.

Keywords: Global optimization, Multistart, Hooke and Jeeves, Filter method
PACS: 02.60.Pn

INTRODUCTION

In this paper, we consider solving the challenging nonlinear constrained mixed variable optimization (MVO) problem containing both continuous and integer decision variables

$$\begin{array}{rl} \min & f(x,y) \\ \text{subject to} & g_i(x,y) \le 0,\ i=1,\dots,m \\ & l_x \le x \le u_x,\ l_y \le y \le u_y \end{array} \tag{1}$$

where $x \in \mathbb{R}^{n_x}$, $y \in \mathbb{Z}^{n_y}$ (in particular $y \in \{0,1\}^{n_y}$), n_x is the number of continuous variables, n_y is the number of discrete variables (integer or binary), l_x and l_y are the vectors of the lower bounds for the continuous and discrete variables respectively, and u_x and u_y are the vectors of the corresponding upper bounds. We will use n to represent the total number of variables of the problem. There are two types of approaches to solve this kind of problem: deterministic and stochastic methods which include most of the well-known metaheuristics. The most common deterministic approaches for solving MVO problems are branch and bound techniques, outer approximation, general Benders decomposition and extended cutting plane methods. The main advantage of these approaches is that they may guarantee to find the global optimum, although they also require large amounts of computational time. Deterministic methods suffer from the problem of dimensionality. In general, the complexity rises exponentially with the dimension of the problem. On the other hand, stochastic approaches for global optimality are common and easy to implement. Their convergence studies usually prove that the global optimum will be obtained in infinite time with probability one. One of the most widely used stochastic algorithms is the so called multistart. It is a popular algorithm due to its simplicity and wide applications. Genetic and evolutionary algorithms for solving problems like (1) are presented in [1, 2]. Known swarm intelligence based optimization algorithms, such as ant colony optimization, have been used to solve MVO problems [3]. Heuristics are also common. For example, in [4], an effective exhaustive enumeration method where only a portion of all candidate suboptimal solutions realized during the search are examined and poor points are discarded in favour of more promising ones, is proposed.

Recent derivative-free methods for locating local minimizers of MVO problems are presented in [5, 6]. In the first paper, the generalized pattern search algorithm for linearly constrained (continuous) optimization was extended to mixed variable problems and the constraints are treated by the extreme barrier approach. The second performs a minimization of a penalty function distributed along all the variables. Continuous and discrete search procedures, as well as a penalty parameter updating rule are the main parts of the presented method.

In this paper, to globally solve MVO problems, a multistart method that uses an extended Hooke and Jeeves (HJ) coupled with a filter based method as a local search procedure, is presented. Although the concepts of exploratory and pattern moves of the original HJ method [7, 8] are used, extensions to handle mixed integer variables and inequality constraints throughout the filter set methodology [9], are included.

In the next section, we present the multistart HJ filter based method for solving MVO problems like (1) and in the last two sections we show the numerical results and conclude the paper.

MULTISTART HOOKE AND JEEVES FILTER BASED METHOD

Multistart methods are very popular when globally optimal solutions are mandatory. Using a classical nonlinear local solver, the basic idea is to repeatedly run the local solver starting from randomly selected points, reaching different optimal solutions. Local and global solutions found during the search process are stored and the best is selected as the proposed globally optimal solution. A multistart method has a limited guarantee that it will converge in probability to a global optimal solution, i.e., the probability that the globally optimal solution will be found tends to one as the number of runs of the local solver increases.

The herein implemented multistart HJ filter algorithm uses uniformly distributed starting points within the bounds $[l_x, u_x]$ and $[l_y, u_y]$ to find multiple minima, including the global minimum. The numbers randomly generated inside $[l_y, u_y]$ are rounded to the nearest integer.

The local solver is a derivative-free pattern search method that is prepared to handle inequality constraints by means of a filter methodology [1, 9]. The main steps of each iteration of the multistart algorithm are:

Step 1 Generate a starting point (x, y);
Step 2 Produce a minimizer (x^+, y^+), running a local solver $\mathscr{L}$ starting from (x, y);
Step 3 Check stopping conditions.

The stopping conditions aim to find a solution within an error of ε_{tol} or an approximation when the number of function evaluations reaches $Nfe_{\max}$.

The local solver $\mathscr{L}$ herein proposed is a Hooke and Jeeves filter (HJ-Filter) based algorithm. The extensions consider two crucial approaches. One is related with the definition of the pattern in the HJ search procedure to take care of continuous as well as discrete variables. The other uses the filter methodology to handle the inequality constraints.

The local search procedure is an iterative method that is applied to a sampled point (x, y) and provides a trial point (x^+, y^+) that is an approximate minimizer of the problem (1).

Using a filter methodology [9], the basic idea is to interpret (1) as a bi-objective optimization problem aiming to minimize both the objective function $f(x, y)$ and the nonnegative constraint violation function $\theta(x, y) = \|g(x, y)_+\|_2^2$, where $v_+ = \max\{0, v\}$. The filter technique incorporates the concept of nondominance, borrowed from the field of multi-objective optimization, to build a filter set that is able to accept trial approximations if they improve the constraint violation or objective function value. A filter $\mathscr{F}$ is a finite set of points (x, y), corresponding to pairs $(\theta(x, y), f(x, y))$, none of which is dominated by any of the others. A point (x, y) is said to dominate a point (x', y') if and only if $\theta(x, y) \leq \theta(x', y')$ and $f(x, y) \leq f(x', y')$.

A rough outline of the HJ-Filter method is the following. At the beginning of the iterative process, the filter is initialized to $\mathscr{F} = \{(\theta, f) : \theta \geq \theta_{\max}\}$, where $\theta_{\max} > 0$ is an upper bound on the acceptable constraint violation.

The search begins with a central point, the current approximation $(\bar{x}, \bar{y}) \leftarrow (x, y)$, and defines a sequence of at least n trial approximations along, first the positive and then the negative, of the unit coordinate vectors $e_i \in \mathbb{R}^{n_x}$ and $e'_i \in \mathbb{R}^{n_y}$, with a fixed step size $\alpha_x \in (0, 1]$, as follows:

$$x^+ = \bar{x} \pm \alpha_x D e_i \ \text{ and } \ y^+ = \bar{y} \pm e'_i, \ i = 1, \ldots, n, \tag{2}$$

where $D \in \mathbb{R}^{n_x \times n_x}$ is a weighting diagonal matrix. First, when (x^+, y^+) falls outside $[L, U]$, where $L = (l_x, l_y)^T$ and $U = (u_x, u_y)^T$, the point is projected onto the search space $[L, U]$. Then, each time a trial point is found that improves over $(\bar{x}, \bar{y})$, reducing θ or f by a certain amount (see (3) below), and is acceptable by the filter, the (x^+, y^+) is accepted and replaces $(\bar{x}, \bar{y})$.

When the search along the n coordinate vectors terminates, the most nearly feasible point (it may be a feasible one) among the accepted trial points is selected. Let that point be denoted by (x^{inf}, y^{inf}). If $(x^{inf}, y^{inf}) \neq (x, y)$, the search is successful and the vector $(x^{inf}, y^{inf}) - (x, y)$ defines a promising direction, known as pattern direction; otherwise the search is considered unsuccessful, and a restoration phase is invoked.

When a successful search occurs, a move along the pattern direction is carried out. The sequence of searches along the n coordinate vectors, as shown in (2), are conducted but with $(x^{inf}, y^{inf}) + ((x^{inf}, y^{inf}) - (x, y))$ substituted for $(\bar{x}, \bar{y})$.

At the end, the most nearly feasible point among the generated trials is selected. When a new acceptable (x^{inf}, y^{inf}) is found then it is accepted as the new iterate, replaces $(\bar{x}, \bar{y})$, and the pattern move is repeated. If, on the other hand, all possible trial approximations (x^+, y^+) are dominated by the current filter, then all are rejected, and the search returns to the previous selected iterate.

To avoid the acceptance of a point (x^+, y^+) in (2), or the corresponding pair $(\theta(x^+, y^+), f(x^+, y^+))$, that is arbitrary close to the boundary of $\mathscr{F}$, the trial (x^+, y^+) is considered to improve over $(\bar{x}, \bar{y})$ if one of the conditions

$$\theta(x^+, y^+) < (1 - \gamma_\theta)\, \theta(\bar{x}, \bar{y}) \ \text{ or } \ f(x^+, y^+) \leq \left(1 - \gamma_f\right) f(\bar{x}, \bar{y}) \tag{3}$$

holds, for fixed constants $\gamma_\theta, \gamma_f \in (0,1)$. We note that if a sequence of trial points is feasible, the condition (3) guarantees that the trial approximation (x^+, y^+) must satisfy the second condition in order to be acceptable. This way the optimal solution is guaranteed. We also note that whenever a point is acceptable, the point is added to the filter $\mathscr{F}$, and all dominated points are removed from the filter.

When it is not possible to find a non-dominated trial approximation, a restoration phase is invoked. In this phase, the most nearly feasible point in the filter, $(x_{\mathscr{F}}^{inf}, y_{\mathscr{F}}^{inf})$, is recuperated and the searches along the n coordinate vectors, as shown in (2), is repeated, but with $(x_{\mathscr{F}}^{inf}, y_{\mathscr{F}}^{inf})$ substituted for $(\bar{x}, \bar{y})$. If a non-dominated trial approximation is found, this point becomes the central point of the next iteration. Otherwise, the iteration is unsuccessful, the search returns back to the current $(\bar{x}, \bar{y})$, the step size α_x is reduced, and a new search, as shown in (2), is repeated about $(\bar{x}, \bar{y})$. When α_x falls below $\alpha_{\min}$, a small positive tolerance, the search terminates since first-order convergence has been attained [8].

NUMERICAL RESULTS

This section aims to analyze the performance of the presented extended HJ-Filter when integrated as the local solver into the `Multistart` function from MATLAB™ Optimization Toolbox, to solve a set of 12 MVO benchmark problems (f_1 – f_{12}), fully described in the Appendix of the paper [1]. From here now, this version will be denoted by M+HJ-Filter. We also present a comparison with the filter-based genetic algorithm (FGA) in [1] and the hybridized ant colony algorithm with a local search (Acomi$_2$) in [3]. The parameters of the M+HJ-Filter algorithm are set after an empirical study as follows. The number of randomly generated starting points in the multistart is 10 and the parameters for the stopping conditions are $\varepsilon_{tol} = 10^{-3}$ and $Nfe_{\max} = 10000$. In the local solver, we set $\gamma_\theta = \gamma_f = 10^{-8}$, $\alpha_{\min} = 10^{-4}$ and $\theta_{\max} = 10^2 \max\{1, \theta(\bar{x}, \bar{y})\}$. A solution is considered feasible when $\theta(x, y) \leq 10^{-8}$.

Each problem was solved 30 times and the average of the best results are reported. Table 1 lists the known global optimal solution, 'f^*', the average of the *best* objective function values (over the 30 runs), 'f_{avg}', the average number of function evaluations, 'Nfe_{avg}', and the success rate, 'SR', in %. For each run, the *best* objective function value is identified as the feasible point that has the least function value among all the located local and global optimizers. In the `Multistart` function, a new optimizer is identified by checking its objective function value and the vector itself with those of previously located optimal solutions. A tolerance of order 10^{-6} is used in the relative difference of f and of (x, y). Furthermore, a run is considered to be successful if the best obtained solution has an error of ε_{tol} relative to f^*. For a fair comparison, ε_{tol} is set to 10^{-4} when a comparison is made with [1, 3] on the problems f_3, f_6, f_7, f_9, f_{10} and f_{11}. The character '–' in the table means that the required information is not available in the cited paper. Bold values show the best obtained results by M+HJ-Filter, FGA and Acomi$_2$. Compared with FGA and Acomi$_2$, the presented M+HJ-Filter algorithm for MVO problems is both effective and efficient for solving the reported test problems.

CONCLUSIONS

We presented a multistart approach that uses an extended version of the Hooke and Jeeves, coupled with a filter based method as a local solver for solving MVO problems. The proposed HJ-Filter algorithm relies on a simple heuristic to be able to handle integer and/or binary variables, and the inequality constraints are handled by the popular filter set methodology. The integration of the HJ-Filter solver into the `Multistart` function turned out to be very effective. Furthermore, neither analytic nor numerical derivatives are required in the proposal. The new algorithm has been tested with benchmark problems and compared with two stochastic methods, one is a recent filter-based genetic algorithm and the other relies on the ant colony swarm approach. From the comparison, we conclude that the

TABLE 1. Comparison of M+HJ-Filter with FGA [1] and Acomi_2 [3]

				M+HJ-Filter			FGA			Acomi_2		
Prob.	n_x	n_y	f^*	f_{avg}	Nfe_{avg}	SR	f_{avg}	Nfe_{avg}	SR	f_{avg}	Nfe_{avg}	SR
f_1	1	1	2.00000	**2.0000**	391	100	2.0005	440	100	–	–	–
f_2	1	1	2.124	**2.1245**	**1150**	100	2.1245	1769	100	–	–	–
f_3	2	1	1.07654	**1.0767**	**3855**	100	1.0767	8074	100	1.1459	4250	60
f_4	2	1	99.24521	**99.2401**	**156**	100	99.2401	1225	100	–	–	–
f_5	3	4	3.55746	**3.5580**	4517	100	3.8956	10172	42	–	–	–
f_6	3	4	4.5796	4.5797	5443	97	5.1322	8125	53.3	**4.5796**	731	100
f_7	1	1	-17.0000	-16.9997	1214	100	-17.0000	999	100	**-17.0000**	**307**	100
f_8	3	2	-32217.4	**-32217.4**	180	100	-32217	6053	100	–	–	–
f_9	2	3	7.66718	7.6847	2492	100	7.7406	4720	80	**7.6672**	363	100
f_{10}	1	1	-2.444	**-2.4444**	**185**	100	-2.4444	230	100	-2.4444	270	100
f_{11}	1	2	3.2361	**3.2362**	1488	100	3.4208	5616	50	23.475	1180	0
f_{12}	1	1	1.125	**1.1250**	286	100	1.1256	428	100	–	–	–

herein presented multistart HJ-Filter method behaves rather well and performs better than the other two in comparison. Future developments include some parameter testing and a convergence study of the algorithm when solving real world applications, namely those concerned with water or transport network design.

ACKNOWLEDGMENTS

This work was financed by FEDER funds through COMPETE (Operational Programme Thematic Factors of Competitiveness) and by portuguese funds through FCT (Foundation for Science and Technology) within the projects FCOMP-01-0124-FEDER-022674 and PEst-C/MAT/UI0013/2011.

REFERENCES

1. A. Hedar and A. Fahim, Filter-based genetic algorithm for mixed variable programming, *Numerical Algebra, Control and Optimization*, 1(1), 99–116 (2011).
2. Y. C. Lin and K. S. Hwang, A mixed-coding scheme of evolutionary algorithms to solve mixed-integer nonlinear programming problems, *Computers and Mathematics with Applications*, 47, 1295–1307 (2004).
3. M. Schlüter, J.A. Egea, and J.R. Banga, Extended ant colony optimization for non-convex mixed integer nonlinear programming, *Computers and Operations Research*, 36, 2217–2229 (2009).
4. V. K. Srivastava and A. Fahim, An optimization method for solving mixed discrete-continuous programming problems, *Computers and Mathematics with Applications*, 53, 1481–1491 (2007).
5. M.A. Abramson, C. Audet, J.W. Chrissis, and J.G. Walston, Mesh adaptive direct search algorithms for mixed variable optimization, *Optimization Letters*, 3, 35–47 (2009).
6. G. Liuzzi, S. Lucidi, and F. Rinaldi, *Derivative-free methods for constrained mixed-integer optimization*, Report R. 11-11, Istituto di Analisi dei Sistemi ed Informatica “Antonio Ruberti”, CNR, Italy (2011).
7. R. Hooke and T.A. Jeeves, Direct search solution of numerical and statistical problems, *Journal on Associated Computation*, 8, 212–229 (1961).
8. T.G. Kolda, R.M. Lewis, and V. Torczon, Optimization by direct search: new perspectives on some classical and modern methods, *SIAM Review*, 45, 385–482 (2003).
9. R. Fletcher and S. Leyffer, Nonlinear programming without a penalty function, *Mathematical Programming*, 91, 239–269 (2002).