FORUM

Robustness Analysis: Optimisation

Marc Sevaux [a] and Kenneth Sörensen [b]

[a] *University of Valenciennes, France*

[b] *University of Antwerp, Belgium*

Introduction

The discussion developed in this forum is an excellent opportunity to present a different point of view of what we consider to be robustness analysis.

In many real-life combinatorial optimisation problems, robustness is just an important issue as is optimality. However, this aspect of optimisation has long been neglected by researchers. For our purposes, robustness refers to the insensitivity of a solution with respect to the input data. In our opinion, researchers and especially those involved in optimisation should be able to provide solution methods that can find robust solutions.

Definitions

Changing data, uncertainty, and dynamic modifications of data lead current optimisation methods to poor solution. One way to deal with stochastic problem data is to find solutions that are *robust*. We distinguish two types of robustness. *Quality robustness* is a property of a solution whose quality, measured by the objective function value, does not deviate much from optimality when small changes in the problem data occur.

The second type of robustness is called *solution robustness* and can be described as robustness in the solution space. When changes in the problem data occur, the decision maker might be forced to re-optimise the problem. In this case, the quality of the solution is guaranteed by the optimisation procedure. In some situations however a solution is preferred that is “close” (in the solution space, not the objective function space) to the solution currently used. For example, many manufacturers operate with a production schedule that repeats itself on a regular basis (e.g. daily or weekly). When e.g. a new job needs to be scheduled, the problem is re-optimised, but the new production schedule should be as similar as possible to the one currently used.

This type of robustness stresses the importance of solution stability. The two types of robustness are not entirely equivalent in the sense that quality robustness is a property of a solution that is insensitive to changes in the problem data *before* these changes occur, whereas solution robustness refers to the stability of a solution *after* changes have occurred.

A simple framework for robust optimisation

Consider an optimisation problem for which the data are uncertain or stochastic. If we want to find an optimal solution, we need to adapt a search procedure to be able to take into account the stochastic nature of our problem.

Based on our experiments, we argue that meta-heuristics can very easily be adapted to the requirements of a stochastic problem formulation. That there is a need for robust meta-heuristic optimisation is recognised in the influential book “Robust discrete optimisation” [KOU97], when the authors say on p. 354: “*We believe that* *considerable more effort should be spent in systematic development of [...] metaheuristic frameworks, which with minimal adjustment effort can be applied to a large class of robust optimisation problems [...] ".*

In metaheuristics, the search towards an optimal (or near-optimal) solution is guided by successive evaluation of solution in a sequential mode (simulated annealing, tabu search, …) or in a parallel way (genetic algorithm, ant colony optimisation, …). If we are able to take into account the stochastic nature of the problem at this step, we should be able to guide the search towards a robust solution. Hence, one way to do it is to modify the evaluation function and evaluate the robustness of the solution at this step. This is done by replacing the evaluation function by a so-called robust evaluation function.

Quality robustness

Let *x* be a solution of an optimisation problem. The quality of *x* is computed by an evaluation function *f(x)*. When we want to indicate that *f* has parameters, we write *f(x,P)*, where *P* is the set of problem data. To allow the robust algorithm (RA) to find robust solutions, the evaluation function *f(x)* is replaced by a *robust evaluation function f*(x)*. The robust evaluation function for quality robust solutions adheres to the following principles [SOR01,SOR03]:

Principle 1: Each solution is implemented on a modified set of characteristics $S_i(P)$. *S* is a *sampling function*, that takes a random sample from the stochastic elements of *P*. $S_i(P)$ is the *i*-th set of sampled parameters of *P*. We call the implementation of a solution on a modified set of characteristics a *derived solution*.

Principle 2: Several evaluations of a solution *x* on a sample of *P* are combined into a new evaluation function. An evaluation of a derived solution is called a *derived evaluation*. This new function is the *robust evaluation function f*(x)*.

A possible form of a robust evaluation function is a weighted average of *m* derived evaluations:

$$f^*(x) = \frac{1}{m}\sum_{i=1}^{m} c_i \; f(x,S_i(P))$$

where c_i is a weight associated to this derived evaluation according to its importance and *m* is the number of derived solutions to evaluate.

A more conservative robust evaluation function may examine the worst-case performance of a solution across all derived evaluations:

$$f^*(x) = Max(i=1 \cdots m,\; f(x,S_i(P)))$$

if *f* has to be minimised.

Solution robustness

Solution robustness is a property of a solution that is similar to a given baseline solution, x_0 *i.e.* for which the *distance* to the baseline solution (as measured by some distance function) is small. Of course, solution robustness cannot be used as the only objective, since solution quality or quality robustness should always be taken into account. The need for solution robustness therefore automatically transforms the problem into a multi-objective one and a solution should be found that simultaneously has a high quality (robustness) and a small distance to the baseline solution.

In our framework, solution robustness is obtained by measuring the distance between the baseline solution and each solution generated by the metaheuristic along the search. It is assumed that the metaheuristic visits a sufficiently diverse set of solutions, so that at least a fraction of them will be solution robust. A solution is then chosen using a multi-objective decision making process, taking into account the decision maker's preferences for solution robustness and quality (robustness).

A sensible distance measure should accurately reflect the “similarity” between two solutions. The meaning of this concept is highly dependent on the specific situation. For problem where the representation of a solution can be undertaken by a permutation, [SOR03a] provides a set of distance measures based on the *edit* distance (also called distance of Levenstein).

Risk preference

The function $f^*(x)$ estimates the average performance or the worst case performance of a solution, given that some of the parameters of the problem are stochastic. Clearly, the worst case performance measure will lead to solutions that are more conservative. Solutions found using this form of the robust evaluation function will hedge only against the worst possible incidence, independent of the probability that this scenario will occur. This type of robust evaluation function can be used by extremely risk-averse decision makers.

A more subtle manner to incorporate the risk preference of the decision maker, is to include into the robust evaluation function an estimate of the probability that the quality of a solution will deviate from its expected value. A possible measure is the standard deviation of the quality of a given solution over all samples: This could be done easily along the search and keep as a second attribute of the solution.

The two measures (the robust evaluation and the standard deviation computed) can be integrated in a multi-objective decision making approach. A possible way is to find the solution that minimises $f^*(x) + \gamma^*(x)\sigma.$, where γ is a parameter indicating the risk-averseness of the decision maker. A more advanced way is to retain all efficient solutions and choose one according to a multi-objective decision making method.

Application in scheduling and routing problems

To demonstrate the efficiency of such type of approach, we tackle two types of problem. The first one considers the optimisation under uncertainty of a one-machine scheduling problem presented in [SEV04]. The metaheuristic developed for this problem is a genetic algorithm. The second type of application is dedicated to the robust and flexible vehicle routing problem presented in [SOR04]. For this latter application, the metaheuristic is a memetic algorithm in which the population is carefully managed. This technique is called MA/PM memetic algorithm with population management.

Conclusion

To conclude, we can summarize our approach as follow. Based on an existing metaheuristic, the stochastic nature of the problem can be taken into account through a robust evaluation function that replaces the standard evaluation function and guides the search towards a robust solution. This approach has been applied successfully to two types of applications, routing and scheduling that can be considered as two major categories of combinatorial optimisation problems.

References

[KOU97] P. Kouvelis and G. Yu, 1997. *Robust discrete optimisation and its applications.* Kluwer academic publishers, Dordrecht.

[SEV04] M. Sevaux and K. Sörensen, 2004. A genetic algorithm for robust schedules in a one-machine environment with ready times and due dates. *4OR* 2(2), pp 129-147.

[SOR01] K. Sörensen, 2001. Tabu searching for robust solutions. In *proceedings of the 4th metaheuristics international conference.* Porto, Portugal, pp 707-712.

[SOR03] K. Sörensen, 2003. *A framework for robust and flexible optimisation using metaheuristics with applications in supply chain design.* PhD. Thesis. University of Antwerp, Belgium.

[SOR03a] K. Sörensen, 2003. Distance measures based on edit distances for permutation-type representations. In A. Barry, editor, *Proceeding of the workshop on analysis and design representations and operators, GECCO conference.* Chicago, USA.

[SOR04] K. Sörensen and M. Sevaux, 2004. Robust and flexible vehicle routing in practical situations. In *proceedings of the 5th triennial symposium on transportation analysis. Le Gosier, Guadeloupe (French West Indies), France.*