Scheduling Countermeasures to Contamination Events by Genetic Algorithms

Marco Gavanelli, Maddalena Nonato, Andrea Peano,
Stefano Alvisi, Marco Franchini

05/07/2013

Letter to editors and reviewers

Dear editors, dear fellow reviewers,

We thank you very much for the insightful comments and suggestions that we got. Thank to your suggestions, we were able to improve very much the paper, that now

- is 25 pages long in two column format
- contains 3 pages of appendices to include all the experimental results; in fact we moved into appendix the box-plots for each scenario in order to improve the readability of the article
- 6 pages in two column format of presentation and discussion of experimental results
- 5 new methods (the Blind Random Search with Constant and Variable speed, and the Local Search) requested by the reviewers, plus 14 methods implemented in previous versions, making a total of 19 methods extensively compared experimentally.
- the techniques explored in this paper span over a wide range of techniques, as diverse as Local Search, Blind Random Search, Genetic algorithms (with 2 different chromosome representations), Hybrid Genetic-MILP algorithm, Deterministic common-sense driven approaches.
- calibration of the size of the population for the genetic algorithms
- constant speed vs variable speed comparison
- 2 levels of hybridization and calibration of the mix level
- statistical significance analysis of the methods, with well-known methods in the statistic literature, including the Friedman test, the Nemenyi post-hoc analysis, and the Bonferroni correction.

The experiments took 231 days of equivalent computation time, plus the human time to implement the various methods, debug them and test them.

We also did many minor improvements and clarifications; a detailed answer to each point raised by the referees follows.

We were allotted six weeks to revise the article, and we did our best to reply to all the issues raised by the referees in such a limited time.

We hope that the reviewers will find the paper a pleasant reading.

Faithfully,
the authors

1 REVIEWER 1

CONTENT Please, answer Yes or No.

Is the paper technically sound? Yes
Is the work described original? Yes
Does it place new results in appropriate context with earlier work? Yes
Does anything need to be added or deleted? Yes
Is the result important enough to warrant publication? Yes
Are the results properly supported by experiments? No
Is the experimental analysis significant and correct? No

1.1 Detailed comments to authors:

The article addresses a very original problem consisting in scheduling the activation of devices on the water distribution networks in order to minimize the impact of some contamination event. If the decision problem in itself is well-known (it is a multiple TSP), the objective function used in the present problem (minimize the consumed contaminated water volume) is very complex to model and can only be estimated by running an expensive simulation.

The description of the problem and its context is clear.

I think a number of aspects should be clarified and improved in the paper:

Q1 Some details are missing in the description of the proposed approach, in particular for the case of the encodings based on activation times (the main technique introduced in the paper), it is never clearly said how the initial population is computed. One can guess that it is based on a random sampling of the possible activation times followed by a feasibility restoration (FRP procedure) of each individual but it is not explicitly said. Furthermore, in the context of an encoding based on activation times, it seems that there should be a notion of parameter representing the "horizon" of the schedule, all activation times being constrained to be smaller than this horizon. I suppose the selection of a value for this horizon has an impact on the performances of the approach. Something should be said about it.

A1 Regarding population generation, all algorithms start from the same population so that comparison is not biased by different initial populations. At the end of section 4 of the previous version we said that "Other features common to all the GA families further introduced are: $(\cdots)$ and random generation of the initial population. In particular, the initial population is generated as follows. Each gene is created by picking at random a device identifier, according to a uniform distribution. If the selected device already appears in the chromosome, it is discarded and the process is repeated. Then, a team identifier is randomly selected, and it is assigned to the device, guaranteeing that at least one device is assigned to each team. For the variable speed case, a third vector of n elements is created by sampling according to a uniform distribution the discrete set $\{0, \ldots, U\}$ of feasible pause durations. As it will be clear in the following, this representation is based on the Two Chromosome encoding; the translation of the initial population into the other encodings is straightforward."

In the new version of the paper, we also recall at the end of Section 7 how the initial population is built.

Regarding the time horizon, we thank the Reviewer for raising this point which is now clarified in the paper. Now it is well distinguished between the scheduling horizon and the simulation horizon. The latter must encompass the whole period during which the contaminant is present in relevant concentration at any point of the network. Therefore, as now explained in the paper (Section 2), the simulation horizon must span several hours in order to provide a precise evaluation of the objective function value. The former, on the contrary, is much shorter. In fact, when devices are activated long after the alarm is raised, high concentration of the contaminant have been present in the consumed water for a long time, so that the solution values are very poor, indeed close to the "no action" option ones. Therefore, there is no point in searching solutions with high activation times": although feasible, they are poor quality solutions. In the variable speed variant we implement this fact by setting an upper bound on the duration of pauses. This is now made clear in Section 2. In that section we also give further explanations about the possible inter-relations among different devices' activations and the potential advantages of scheduling with pauses.

Q2 - It is mentioned on several occasions in the paper that the proposed approach performs better than usual ”commonsense inspired criteria”. It would be useful to provided some evidence of that in the experimental section.

A2 We considered two types of ”commonsense inspired criteria”: one aims at ending the whole process of device activation operations as soon as possible, which corresponds to minimizing the *makespan*, i.e., the time at which the latest device is operated; the other seeks to activate each device as soon as possible, which is modeled as the minimization of the sum of the activation times, and it can be seen as a sort of *latency*. In the improved version of the paper we added the solutions obtained by both methods, which have been computed by solving the associated MILP models. It can be noted that both provide poor quality solutions and none of the two improves our solution value for any of the scenarios.

Q3 - The discussion about the pro and cons of using a high-performance MILP solver (p16) is weird. The bottom line is that if the MILP solver is too good and always finds the optimal solution, this can damage the diversity of the approach and thus, it is better to stop the search earlier by using an optimality tolerance as it will provide some kind of random effect on the generated individuals. I think that if this explanation is right, then it would be better to make this randomness effect explicit in order to better control it. For instance, you could randomly not consider some coordinate(s) j in the objective function that measures the distance to the parents.

A3 The issue of population diversity raises near the end of the search when the population diversity is low, and many individuals differ by a couple of coordinates (e.g., 1 or 2 coordinates). In MILPX any combination of the activation times of the two parents is a vertex of the hypercube and it is quite likely that at least one such vertex is a feasible solution and already belongs to the population. Using an ϵ tolerance for optimality allows the solver to pick any feasible solution in the ball of ray ϵ centered at a vertex, increasing the chance that such a solution does not belong to the current population. We believe that the ϵ parameter provides a way to explicitly control the randomness of the choice of MILP solver. We tried to clarify this issue. On the other hand, the suggestion of the Reviewer (not considering some coordinates when computing the distance) does not provide a way to control the randomness, since the real distance of the new point can not be bounded a priori. In fact, the MILP solver would compute a minimal distance from parents only for a subset of coordinates, while the others can span over the whole set of feasible combinations, following a non-deterministic behaviour.

May be the value of ϵ could be tuned along the search, according to the percentage of clones generated. However, as we already pointed out, the time we have been allotted for the revision does not allow us to test a new version. A comprehensive experimentation of this new version would take about 578 days of equivalent computation time.

Q4 - The justification for the use of a GA approach for the problem is that it can explore wide parts of the search space with a limited number of solution evaluations which indeed makes sense in the context of the application. A GA will be efficient if, within the limited number of solution evaluations, it is able to capture some building blocks of the problem. The experimental evaluation does not give enough evidence that the GA is able to find those building blocks. In particular, the GAs should be compared with a simple random sampling of the solution space. If there is a limit of 500 evaluations, why not just randomly draw 500 random solutions in the solution space (for instance 500 random vectors of activation times made feasible using FRP) and evaluate them. Are the GA variants significantly better than random sampling? A related question is the population size (question is related because random sampling is more or less equivalent to a population size of 500 for which no GA operators are applied). It is said on p15 that the initial population size of 20 was selected based on the only $2C^{CS}$ encoding. Usually the best population size highly depend on the encoding, so the best population size for the H encodings may be quite different. Did you make some experiments varying the population size for activation times encodings? By the way, for $2C^{CS}$, the efficiency seems to systematically increase when decreasing the population size on $[20, 100]$. Did you try values smaller than 20?

A4 We believe that the activation times of (some of) the devices are the building blocks of good solutions, and for this reason the time based encoding is better performing. However, we followed the advice of the Reviewer and performed a blind search realized by 500 random sampling on both the constant and the variable speed search spaces. For each scenario, they were repeated 100 times so that a fair comparison is possible. For each scenario, the Wilcoxon Signed-Ranks test tells that the worst GA (the 2-Part GA) performs *significantly* better than the blind search, and we now mention this in Section 8. Moreover, in order to further justify the use of GAs, we also implemented a basic Local Search, whose neighbourood is based on moving one device from its current position to right after any other device. In order to save function evaluations we implemented a first improvement strategy. Nevertheless, results confirm that with such a limited number of allowed evaluations, neighbourood based strategies are not promising (see Section 8).

Finally, as far as the population size calibration is concerned, this parameter value is strongly dependent on the number of allowed function evaluations. In this application, the hydraulic engineers set this value to 500. Population size is now discussed in Section 8, and we present in Figure 4 the solution quality for the $2C^{CS}$ algorithm for a population size from 10 to 100. It can be noted that the ♯10 configuration converges quickly but to a higher level w.r.t. the ♯20 configuration. While the population size has not been calibrated for the other encodings, the time-based encoding, which now appears to dominate the others, could only take advantage from an ad-hoc population size.

Q5 - I think that some effort should still be made to better justify the proposed approach. The experimental section, although it has been extended, does not make it clear if and why the proposed GA is better than other (possibly simpler) approaches such as a pure random sampling. By the way, I did not find the letter explaining how the reviewers comments were taken into account.

A5 Pure random sampling and local search methods are now deeply discussed in Section 8. We also added further considerations about common sense schedulings, such as the so called “Makespan” and “Latency” solutions. Results clearly suggest that Hybrid GAs outperform all the above methods.

2 REVIEWER 2

This paper proposes three genetic algorithms (GA) for an optimization-simulation problem (i.e. an optimization problem for which the objective function is obtained through simulation) arising from a countermeasure scheduling problem in the context of a water distribution network contamination disaster.

We note that this paper is an extended version of a paper presented in an LNCS issue on Evolutionary Computation (proceedings of EvoCOP 2012.), mainly including additional experiments and discussions.

In Section 2, The need for using metaheuristics and evolutionary algorithms is well-justified by the absence of lower bounds, poor quality of existing feasible solutions (upper bounds), and absence of cost-relevant neighborhood structure for local search.

Section 3 details the relevant literature, and, in particular, discusses the related multiple traveling salesman problems and parallel machine problems and the commonly encountered encodings of these problems in GA.

Section 4 provides a high level description of the GA/Simulation framework for the considered problem.

In Sections 5,6,7 the proposed GA are detailed in terms of encoding scheme, crossover, mutation, symmetry considerations and inclusion of variable speed characteristics.

Sections 5 and 6 consider sequence-based encodings that work only on feasible solutions but include, as a counterpart, a lot of redundancy.

Section 7 presents a direct start time vector encoding which by definition is redundancy-free but may represent infeasible solutions. The authors propose a mixed integer linear programming model to restore feasibility from an infeasible vector by computing its closest feasible vector according to the L1 norm. The problem is shown NP-hard by equivalence to a parallel machine problem with earliness and tardiness penalties (as the infeasible vector may be described as due dates for the new vector). A binary crossover is defined, possibly issuing an infeasible vector and repairing it with the above-defined MILP. A MILP-based crossover is also proposed using the principle of generating a solution close to the hyper-parallelepiped defined by the two parent vectors.

Section 8 presents highly detailed analysis of the experimental results. Among others, there is an interesting discussion on how allowing suboptimality in the MILP solver may produce more diversified results for the MILP-based crossover. A statistical analysis confirms that the MILP based GA outperforms the sequence-based GAs. This is also due to the fact that thigh CPU times are anyway required by the simulation part, which makes it worth spending a little more time in MILP solving.

2.1 Detailed remarks:

Q1 - To summarize, this paper presents an efficient genetic algorithm hybridized with MILP solving and simulation to solve a practical scheduling problem in the context of water contamination disaster. I appreciated that the use of the proposed method is particularly well-justified which is not always the case when GA come into play. I would suggest to reduce the experimental result section and especially the statistical analysis part. No such developments are needed to reach the conclusion. Subject to this minor remark, I suggest to accept this paper.

A1 We moved the final boxplots in the appendix to make the paper lighter. We believe the statistical analysis is necessary when the experimental phase involves a huge degree of randomness.

Q2 - Sentences should not start with a numbered reference e.g. P.6 C1 a sentence starts with "[27] tackles the parallel machine [...]". Use author names first.

A2 Thank you for your suggestion. We updated the citations.

Q3 - Section 5 and section 6 contain only one subsection

A3 We have deleted the indexes from the titles of the two subsections.