Universität Hamburg
Department Informatik
Knowledge Technology, WTM

Examination of a framework for multi-objective analysis of computational models

Seminar Paper

Knowledge Processing

Jan Fabian Schmid
Matr.Nr. 6440383
2schmid@informatik.uni-hamburg.de

10.11.2015

Abstract

Your text here...

Contents

1 Introduction

1.1 Problem

The studied paper [1] tries to cover the problems arising at the use of sophisticated computational models. In more and more areas computational models are used for increasingly hard scientific issues. Doncieux et al. break the problem down to the epistemic opacity. Which they define as follows:

A process is epistemically opaque relative to a cognitive agent X at time t just in case X does not know at t all of the epistemically relevant elements of the process.

Thus epistemic opacity is a term to describe the state of knowledge of an agent in matters of a specific process at a specific point in time. It says that the agent has not recognized the process in its entirety at the point in time. In case of computational methods the term can be used to describe a situation in which a scientist developed a complex model, to study an even more complex process, during the simulation of the model he is however not aware which model parameters are interesting to use since the model is too complex to know the implications of different parameter values. This holds for the time until he finally studied the model in an extent that allows him to predict the effects of changed parameter values.

Epistemic opacity may stem from a complicated mathematical foundation to the model, from the complex interaction of different model parts with each other or from the effect of different input-parameters. They say: „[...], the computational model is in itself a complex system to study in order to unravel its unforeseen features“ [1] p. 3. This problem can lead to an extensive search in parameter space for the „best“ parameter values. The researcher might have an intuition for some good parameter values, but these are biased by his assumptions about the model.

Furthermore in a complex system it might not be obvious what properties of the model are actually desirable. Hence the „best“ model parameters wouldn't be recognized if found. Since the model should lead to novel knowledge, and proven or disproven assumptions about the modelled system, interesting parameter-sets for further studying are desired. Interesting parameters should be somehow different from other parameters and state something about the model.

To overcome the epistemic opacity the researcher would have to analyse the complex model by approximating it with a model again (cf. [1] p. 3). This process of modelling a modell could be repeated multiple times, each step using a more abstract and simple model, until it is easy enough to understand by the researcher. The knowledge gained from the simpler models could then be used to understand the more complex models. At some point the problem is understood well enough to use the primary model with confident insight.

1.2 Suggested solution

Because modelling a model becomes exponentially intricate and success isn't guaranteed it can hardly be a feasible approach to study a problem. The paper suggests to exploit the fact, that the computational model can be computed in a „huge number of experiments“ [1] p. 3. They suggest a framework method that once set up searches automatically for interesting model parameters, which can then be analysed. The framework requires that a set of functions that describe the model with three properties are given:

1. The functions measure the model performance.
2. For optimal model performance the functions have to be maximized or minimized.
3. Some functions are contrary to each other, as optimizing one function value worsens an other.

The given function set maps parameter space to behaviour space. Calculating many possible solutions allows to find relations between both spaces. Since the performance of individual functions is contrary to each other multi-objective algorithms can be used to find sets of trade-off solutions (cf. [1] p. 3). Doncieux et al. then suggest to use evolutionary multi-objective algorithms, since they are efficient and versatile in usage. Following the steps of the framework interesting parameter sets are thereby found. The paper suggests then some tools to acquire knowledge about the implications of obtained parameter values on model behaviour.

1.3 Goals and structure of this paper

This paper is supposed to give an in-depth analysis of the presented paper [1]. In Chapter 2 some background, necessary to understand the main concepts of the paper, will be given. The proposed frame work of Doncieux et al. will then be presented in a summarized form in Chapter 3. In the following chapter 4 the model and its results will be critically reviewed. Especially the applicability of and assumptions made by the framework will be discussed. Chapter 5 will conclude the paper by stating the achieved insights on the usage and applicability of the framework.

2 Background Information

The crucial concepts to understand the framework for computational methods are presented in this chapter.

At first the topics of Multi-objective problems and what a optimal solution to such an ambiguous problem looks like are outlined. These issues will be discussed in section 2.1. The functionality of evolutionary algorithms is subject to section 2.2. Section 2.3 brings these concepts in context for the usage of evolutionary algorithms in Multi-objective-optimization. And finally in section 2.4 some indicators are introduced, that will be used to analyse the results of multi-objective-optimization.

2.1 Multi-objective optimization

Doncieux et al. [1] formalize optimization problems approximately as follows:

$$\text{Find the parameter } \mathbf{X} = \left\{ \begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_m \end{array} \right\} \in \mathbb{R}^m \text{that maximizes (or minimizes) f}(\mathbf{X}) \text{ under}$$

a given set of constraints.

For $m = 1$ such a task is called mono-objective problem, whereas it's a multi-objective problem for any $m > 1$.

As soon as you want to examine a real world like problem it's very likely that the solution won't be a single one dimensional value. Instead there will be multiple objectives to be considered simultaneously. Fonseca and Fleming state that an optimal solution to a multi dimensional problem is commonly not optimal for each of the objectives (cf. [4] p. 1) This should be the case as long as the different objectives are somehow conflicting. They say further that you would rather aim at a solution that is at least acceptable for all sub-objectives. This however isn't a explicit problem statement. In a non-trivial case there will be several quite different alternative approaches to the problem that are optimal in some aspects. All these approaches should be valid solutions given by a multi-objective optimization-algorithm. To allow this to happen the **Pareto dominance** relation and the **Pareto-optimal set** are introduced. A solution $\mathbf{X_1}$ dominates another solution $\mathbf{X_2}$ if $\mathbf{X_1}$ is not worse than $\mathbf{X_2}$ for any objective and for at least one objective strictly better (cf. [1] p. 6). A solution is called Pareto-optimal if it isn't dominated. All Pareto-optimal solutions form the Pareto-optimal set. Since the Pareto dominance relation is a partial ordering it allows any number of optimal solutions. The Pareto-optimal set contains interesting solutions, because each of its members is a solution that is best in some aspect.

2.2 Evolutionary algorithms

An evolutionary algorithm is a stoachastical approach to obtain a solution to a problem. Because of the stochastic behavior in general the solutions won't be optimal, but are often feasible. Similar to the evolutionary process in nature, a group of individuals is developed over multiple generations under the influence of a pressure to adapt to the environment. An induviduum during this process represents a solution to the problem.

In figure 1 a rough schemata of the important steps of an evolutionary algorithm are shown. At first an (genetic) encoding for all possible solutions to the problem has to be defined in a way that the different functions of the following steps can handle. The first group of solutions has to be initiated randomly.

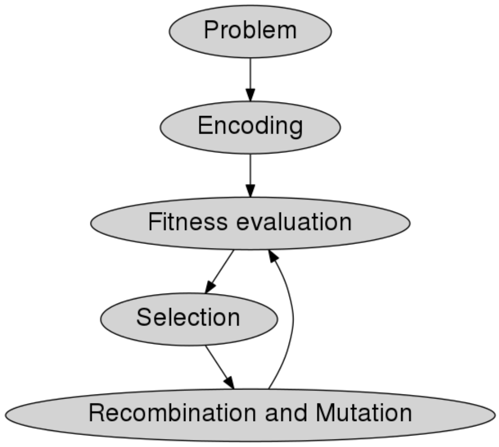

Figure 1: General schemata of an evolutionary algorithm

Now the following steps are used repeatetly until some stopping criterea holds true:

1. **Fitness evalution:** For each individual a **fitness function** evaluates its performance at solving the problem.

2. **Selection**: Select the surviving invividuals, maybe just the best x% of solutions. Some of the survivors are selected as parents for a new generation of individuals. For example a few percent of individuals with the highest fitness score are selected and with probability according to their fitness some more.

3. **Recombination and Mutation:** New solutions are bred out of the selected parents through crossover and mutation.

Such an simple evolutionary algorithm has only local convergence and therefore depends on the random initiation, as it follows a greedy approach, which follows the best solutions in a narrow seach space around the already avaiable solutions.

2.3 Evolutionary Algorithms for Multi-objective optimization

As common optimization-algorithms like gradient descent or simulated annealing aren't suited to deal with multi-objective problems (cf. [4] p. 2) other approaches have been used. One of the strengths evolutionary algorithms have, is to pursue different possible solutions at the same time. That's because in a well structured evolutionary algorithm the surviving individuals aren't only associated to the single best solution at the given time, but also momentary suboptimal solutions are being tracked. Through crossover partially optimal parts of solutions can be exchanged as a whole. Therefore the Pareto front, the current Pareto optimal set during execution of the algorithm, can be explored effectively.

As evolutionary algorithms impose no mathematical constraints on the examined problem. The stochastic approach of evolutionary algorithms however entails to calculate many possible solutions in application to work properly. Also evolutionary algorithms don't ensure to find global optima, instead it converges to some local optimum. Accordingly only an approximation to the Pareto optimal solutions will be found.

The concept used by Doncieux et al. to gain knowledge about a system by analysing the results of a multi-objective optimization algorithm on it is called INNOVIZATION as they state this stands for 'INNOVation through optimIZATION' (cf. [1] p. 5). Evolutionary multi-objective optimization algorithms were already used successfully with this approach as shown in [2], where Efstratiadis and Koutsoyiannis sum up the experience with the INNOVIZTION method for calibration of hydrological models.

2.4 Indicators for multi-objective analysis

To be able to analyse the gained information from multi-objective-optimization on the examined problem Doncieux et al. introduce some indicators (cf. [1] p. 7 et seq.) In the following it is without loss of generality assumed, that all objective-functions are to be maximized to be optimal. The evolutionary multi-objective-optimization is used multiple times on the problem. Each run generates a set of approximations to the Pareto optimal solutions called χ_i for the i-th run. The required indicators are defined as follows.

- The attainment function $\alpha_\chi(z)$ returns the probability to find in the set of results χ **at least one** solution x that attains z, which means here that x dominates z. This function can't be computed directly, but it can be approximated with the empirical attainment function $\alpha_r(z)$ over r sets of approximation results as mean number of sets in which a solution is found that dominates z.

- The attainment surface Ψ_p to an attainment function value p is defined as the hyper surface in behaviour space over points with probability p given by the attainment function. Therefore the 0-attainment surface Ψ_0 covers the set of non-dominated solutions and the 1-attainment surface Ψ_0 represents the worst performing solutions that will certainly be dominated.

- The hypervolume indicator $I_H^p(\chi)$ is defined for a particular reference point p for a set of results χ_0. A simplified attainment function $\beta\chi(z)$ definition is used for the hypervolume indicator, it returns 1 if there is a solution in χ, that dominates z, and 0 otherwise. The hypervolume indicator is then defined as:

  $$I_H^p(\chi_0) = \int_{\psi_p} \beta\chi(z)dz$$

  With ψ_p containing all points, that are equal or greater than p in every dimension. Therefore $I_H^p(\chi)$ measures the area of points above p that are dominated by a solution in χ. In figure 2 a visualisation of the simplified attainment function and its hyper volumne is shown.

- The nadir $\eta(\chi)$ is the point with the least possible value for each objective found in χ. So for each dimension all solutions in χ are scanned for the minimum value. When $\eta^i(\chi)$ is the nadir point to the i-th set of solutions, the conservative nadir point $\bar{\eta}$ is defined over r sets of solutions and its values are the maximum value for each dimension found in any $\eta^i(\chi)$ with $i \in r$. The conservative nadir helps to get rid of outliers and is a more reliable representation of the „worst" returned solution.

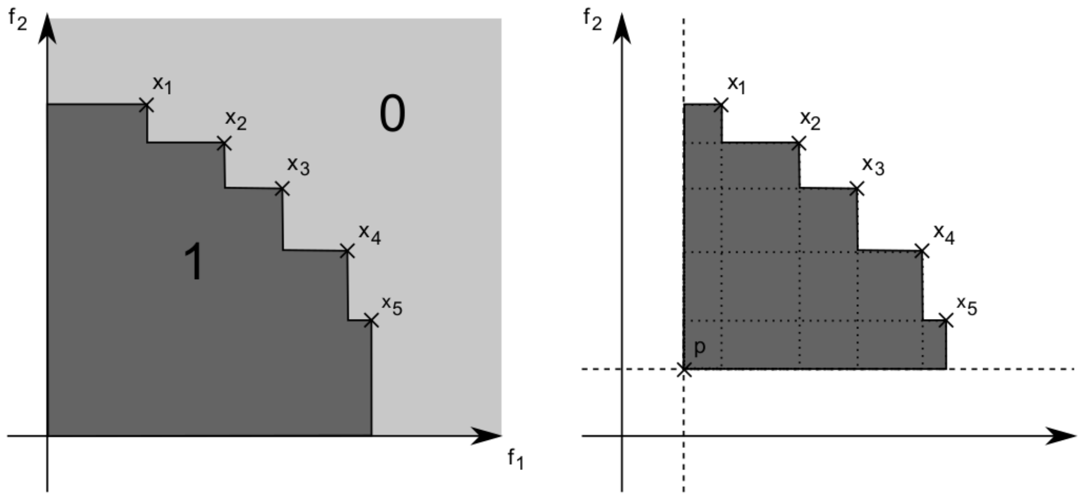

Figure 2: f1 and f2 are the two objective functions in this example. On the left: $\beta\chi(z)$ for $\chi = \{$x1,x2,x3,x4,x5$\}$ and the attainment surfaces Ψ_0 and Ψ_1 (to the simplified attainment function β) are highlighted. On the right: The hypervolumne $I_H^p(\chi)$ is shown. [1]

Summary: *In this chapter the basics concepts used by the framework were introduced. The core part of the framework is a multi-objective-optimization. As shown the evolutionary approach is suitable for this task, its only drawback is the need for calculation of many iterations since there is no global convergence guaranteed. At last a couple of indicators for analysis of the obtained approximations of Pareto-optimal solutions were presented.*

3 Framework description and application

The proposed framework by Doncieux et al. [1] aims at automatically finding parameter values worth to be studied further. There are three steps to application of the framework.

[1] image from [1] p. 8

1. The definition of functions to evaluate the performance of the examined computational method.

2. Obtain an approximation of the Pareto optimal solutions by application of evolutionary multi-objective optimization.

3. Knowledge creation by analysing the solutions.

These steps are specified more in detail in the following sections.

3.1 Search space and objective functions

First the search space as part of the parameter space must be defined. For a set of m unbound parameters $\in \mathbb{R}$ it would be $\mathbb{R}^m$.

The framework relies on the comparison of different Pareto optimal solutions to find interesting parameter values, therefore multiple contrary objective functions have to be defined. They need to be contrary in terms of not linearly dependent, since such a dependency would lead to a set of Pareto optimal solutions with only one member that is optimal for each objective function. A simple second objective for a computational model whose performance actually only depends on a single one might be found in the computational cost. With increasing computation time the algorithm might find better and better solutions. The optimal parameters would then be a compromise between quality of solution and time consumption.

3.2 Application of multi-objective-optimization

If ideally no promising parameter vectors shall be missed it is important to make a dense search in parameter space, otherwise behaviours that only occur in a small area in parameter space might be skipped. As explained in chapter 2.3 evolutionary algorithms are a good choice for the multi-objective-optimization. The local convergence of evolutionary algorithms makes it necessary to run the optimization many times to ensure statistical significance of the results. Therefore the **first step** of application is to repeat the multi-objective-optimization several times. However the results from evolutionary algorithms strongly depend on their parametrisation, it is important to verify the results somehow. Therefore the variability of results to successive runs of evolutionary multi-objective-optimization can be checked. A high variability in results might be a sign of a ill parametrisation, as mentioned by Doncieux et al. [1] (p. 11). Thus the **second step** of application is the evaluation of disparity between multiple iterations of the algorithm. Doncieux et al. propose the following schemata for this:

1. Compute the 0-attainment Ψ_0 and 1-attainment Ψ_1 surface.

2. Calculate the conservative nadir point $\bar{\eta}$ over all independent runs of the optimizer.

3. Compute the hypervolumes of Ψ_0 and Ψ_1 relative to $\bar{\eta}$
   $I_H^{\bar{\eta}}(\Psi_0)$ and $I_H^{\bar{\eta}}(\Psi_1)$ and calculate their difference.

With the difference between these hypervolumes the number of dominated solutions by the best and the worst found solutions is compared. If the difference is high the disparity between the solutions is high and vice versa. This is therfore a measure for the disparity of the worst to the best solutions. If this value is above some threshold step 1 needs to be repeated with different parametrization of the evolutionary algorithm.

For the last step local variability in the approximation of the Pareto optimal set is studied. **Strongly diverging optimal solutions in a small area of parameter space is a sign for a difficult parameter range to explore.** This area of high disparity of solutions might emerge from model instability and a critical point of parameter values might lie there, which should be worth further studying of the researcher. Correspondingly the **third step** to apply multi-objective-optimization is to compute the maximum distance of each point in Ψ_0 to its neighbours.

3.3 Analysis of results

Doncieux et al. provide some assistance for the analysis of obtained approximation to the Pareto optimal set from the previous step. They divide the analysis in three parts.

- Analysis of the results in **behaviour space**. Particular high and low values for objective functions might be interesting. After this a plot of the trend for the objective function values is necessary, to examine the shape of the surface of function values. Thereby relations between objectives can be seen. A funny shaped surface „may even be the symptom of an ill-posed model“ Doncieux et al. [1] (p. 13).

- Analysis of the results in **parameter space**. The relations between model parameters and their objective function values can be interesting. Plotting the trend of a parameter against an objective function might exploit some kind of dependency between them.

- Analysis of the actual proposed **solutions** to parametrization of the model. The Pareto optimal set has to be clustered in multiple groups of similar solutions. This can be done by experts or clustering algorithms. Doncieux et al. suggest to make a classification of the clusters using a decision tree algorithm. The decision tree will find discriminative objective values. These boundary values are best suited to make a decision for each suggested solution as to which cluster they belong to. By examining the constructed decision rules afterwards, one can find out what characterizes the clusters and sets them apart from each other. Each cluster might represent a specific pattern of behaviour.

Summary: *These three major steps define the proposed framework of Doncieux et al. First the problem must be stated and measurements for the success of solutions on it have to be given. Secondly evolutionary algorithms for multi-objective-optimization are applied. And Concluding the obtained Pareto optimal sets are analysed. By following these steps significant and interesting model parameters should be presented to the researcher.*

4 Framework analysis and discussion

Results of the presented experiments
- Were the results expected?
- Positive or negative outcomes?

Usability of the model
- Effort and benefits of using it

Applicability (research question)
- What assumptions and constraints are made on the computational model?
- What are fields of application for the framework? (Is it everywhere applicable?)

5 Conclusion

How reasonable is the approach of the paper?
Where is it applicable?
future research

References

[1] Stéphane Doncieux, Jean Liénard, Benoît Girard, Mohamed Hamdaoui, and Joël Chaskalovic. Multi-objective analysis of computational models. *arXiv preprint arXiv:1507.06877*, 2015.

[2] Andreas Efstratiadis and Demetris Koutsoyiannis. One decade of multi-objective calibration approaches in hydrological modelling: a review. *Hydrological Sciences Journal–Journal des Sciences Hydrologiques*, 55(1):58–78, 2010.

[3] Agoston E Eiben and JE Smith. Introduction to evolutionary computing. *Assembly Automation*, 24(3):324–324, 2004.

[4] Carlos M Fonseca and Peter J Fleming. An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary computation*, 3(1):1–16, 1995.