JOURNAL OF CLIMATE

Global Optimization of the Analogue Method by Means of Genetic Algorithms

PASCAL HORTON*

University of Lausanne and Terranum, Lausanne, Switzerland

MICHEL JABOYEDOFF

University of Lausanne, Lausanne, Switzerland

CHARLES OBLED

Université de Grenoble-Alpes, LTHE, Grenoble, France

ABSTRACT

The Analogue Method is based on a statistical relationship between synoptic atmospheric variables (predictors) and local weather (predictand), which we aim at forecasting. This relationship is expressed through many parameters that are usually calibrated by means of a semi-automatic sequential procedure. This calibration approach has strong limitations: it is made of successive steps and thus cannot handle parameters dependencies, and it cannot automatically optimize some parameters, such as the selection of the pressure levels and the temporal windows on which the predictors are compared. In order to surpass these limitations, a global optimization technique was assessed, namely genetic algorithms, which can optimize jointly all parameters of the method and get closer to a global optimum by taking into account the parameters dependencies. Moreover, it can choose objectively parameters that were previously manually assessed, and can take into account new degrees of freedom that were unthinkable before. These kind of optimization techniques need however to be tailored to the problem to solve. Multiple combinations of algorithms had to be assessed, and even new operators were developed, such as the *chromosome of adaptive search radius* which was found to be very robust, in order to make recommendations for the use of Genetic Algorithms to optimize the analogue method. These recommendations are the main outcome of this work. It opens new perspective for the improvement of the analogue method, and its application to new regions or to new predictands.

1. Introduction

The Analogue Method (AM) relies on the hypothesis that similar situations in terms of atmospheric circulation are likely to lead to similar local weather (Lorenz 1956, 1969; Duband 1970; Bontron and Obled 2005). The principle consists in sampling a certain number of past situations based on different atmospheric variables (predictors) in order to build a probabilistic forecast for a local weather variable of interest (predictand). The most common usage of the AM is for precipitation forecasting (eg. Guilbaud 1997; Bontron and Obled 2005; Bliefernicht 2010; Marty et al. 2012; Horton et al. 2012; Radanovics et al. 2013; Ben Daoud et al. 2015), but AMs, or equivalent, were also used for short to medium term forecasting of daily temperatures (Radinovic 1975; Woodcock 1980; Kruizinga and Murphy 1983), wind (Gordon 1987), snow avalanches (Obled and Good 1980; Bolognesi 1993), insolation (Bois et al. 1981), and the trajectory of tropical cyclones (Keenan and Woodcock 1981; Sievers et al. 2000; Fraedrich et al. 2003). Applications for monthly forecasts in many countries also exist, including Canada (Shabbar and Knox 1986), Hungary (Toth 1989), the Netherlands (Nap et al. 1981), and England (Murray 1974), as well as seasonal forecasts: Barnett and Preisendorfer (1978), Bergen and Harnack (1982) and Livezey and Barnston (1988).

The purpose of this paper is not to present in details the AM and its different parameterizations, which can be found in Horton et al. (2016a) or Ben Daoud et al. (2015). The concepts one needs to understand here are the following: the AM consists in searching a certain number of past situations in a meteorological archive (typically reanalysis datasets) that are the most similar, according to an analogy criteria, and to extract the observed values of the local weather variable of interest from another archive (typically local or basin averaged time series) in order to build the conditional empirical distribution considered as the probabilistic forecast for the target day. Predictors can be varied: for example the geopotential height at different pressure levels on different temporal windows (time of observation). The method is usually made of several levels of

*Corresponding author address: Terranum, Rue de l'industrie 35 bis, 1030 Bussigny, Switzerland.
E-mail: pascal.horton@alumnil.unil.ch

analogy, leading to successive subsampling on predictors of different nature (eg. atmospheric circulation, moisture variables, vertical motion, and air temperature).

Even though the method is rather simple compared to numerical weather prediction (NWP) models, it contains a certain number of parameters one needs to determine, such as the choice of the predictor variable, its pressure level and temporal window to consider, the spatial domain to use for the comparison, as well as the analogy criteria itself, and finally the number of analogue situations to keep at each subsampling level.

The AM needs to be adapted to every new region considered, because the leading meteorological influences may be specific per region. Even the selection of the pressure levels and the temporal windows should be reconsidered, when not the predictor variable itself. For example, Ben Daoud (2010) found the vertical velocity relevant for the great plains in France, when Horton (2012) found it useless in an Alpine environment, because the vertical motion, which is mainly controlled by orographic effects, was already well related to the atmospheric circulation itself.

Up to now, the optimization of the method has been undertaken by semi-automatic sequential calibration procedures (see Bontron 2004; Horton et al. 2016a, for its definition). However, the selection of predictor variables, pressure levels and temporal windows had still to be made manually before optimizing the spatial windows and the number of analogues. Testing multiple combinations of these is very combinatorial and becomes quickly cumbersome, especially when considering multiple predictors within the same level of analogy. Thus, optimizing the method with the sequential technique is laborious, as many combinations of predictors (variables, pressure levels, temporal windows) have to be assessed. Moreover, proceeding to the optimization sequentially ignores potential dependencies between the parameters of the method, may they be within a single level of analogy or between them. However, such dependencies between them cannot be excluded, which could lead to another parametrization if calibrated together. However, simultaneous calibration of all parameters has never been undertaken so far. Thus, due to the sequential approach, the risk of ending in a local optimum is high and can not be avoided. Indeed, during the AM calibration it has been found that the resulting parameters may vary with initial choices (such as the number of analogues).

When creating the sequential calibration procedure, Bontron (2004) was aware of the problem of dependencies between parameters and wrote: "*We perceive here the combinatorial aspect of our problem: variables and spatial windows are not independent. We will present our results by first searching the best variable [note: e.g. selection of the pressure level and the temporal window for the geopotential height] on a chosen spatial window, and next, the best window for the chosen variable. However, even by repeating the process, are we sure to obtain the optimal combination?*". And later in his work: "*Our approach, which is again to vary the parameters one by one – the others being fixed in a more or less arbitrary manner – may therefore not exactly lead us to the optimal solution*". Bliefernicht (2010) has also faced the combinatorial issue of the parameters of the AM and concludes that one needs to be an expert to have a sense of their respective influence, sensitivity and nonlinear interactions. Ben Daoud (2010), when calibrating the AM, also stated that "*the combinatory aspect related to the calibration was found to be too high for all the parameters to be calibrated simultaneously*".

In order to overcome these limitations, two optimization techniques were assessed. First, Horton (2012) assessed the ability of the Nelder and Mead (1965) method based on a simplex approach. This technique did not provide satisfying results and failed at converging toward a unique solution. The parameter space of the AM is very complex and is inappropriate for a linear optimization technique. The conclusion was that global optimization techniques were necessary in order to calibrate AMs, as it is the only way to optimize all parameters of all analogy levels simultaneously. In addition, it can overcome the systematic manual assessments of all pressure levels and temporal windows. Finally, it can open new perspectives by allowing the addition of new degrees of freedom, such as a weighting of the criteria values between the pressure levels (see Horton et al. 2016b), and the consideration of differentiated spatial windows between the pressure levels. The relevance of Genetic Algorithms (GAs) is presented here, which does not exclude that other global optimization techniques could eventually be successful.

This article is not about discussing the details of the results of an optimization with GAs, but describing how GAs are to be used in order to successfully optimize AMs. Indeed, GAs variants are numerous and always need to be tailored to the problem addressed. This requires intensive and systematic comparisons of operators and options in order to identify the key factors leading the optimization and the respective sensitivity of the options. Such analyses are presented here and will result in recommendations for the use of GAs when applied to AMs. The demonstration of the benefit brought by such an approach on a specific case study is the topic of Horton et al. (2016b).

We will begin by presenting the concepts of GAs as well as the assessed operators (operations performed on the AM parameters; see section 2). The comparative analyses of the operators and their results are presented in section 3, which lead us to the recommendations in section 4.

2. Assessed Genetic Algorithms variants

Genetic Algorithms (GAs) come from the world of stochastic optimization, more specifically from metaheuristic approaches. These are stochastic iterative algorithms that behave like search algorithms by exploiting the characteristics of a problem and are particularly suitable for complex parameter spaces.

GAs are part of the family of Evolutionary Algorithms (Bäck and Schwefel 1993; Schwefel 1993), inspired by some mechanisms of biological evolution, such as reproduction, genetic mutations, chromosomal crossovers, and natural selection. GAs are the most used technique among Evolutionary Algorithms (Bäck and Schwefel 1993), and they are constantly improving (Haupt and Haupt 2004). However, with time, the different methods of Evolutionary Algorithms tend to be similar and share many commonalities (Bäck and Schütz 1996; Haupt and Haupt 2004).

The method was originally developed by Holland (1992) and popularized by Goldberg (1989). Unlike a linear or local optimization, GAs seek the global optimum on a complex surface, theoretically without restriction, but with no guarantee to reach it.

a. Basic concepts of the Genetic Algorithms

GAs mimic the evolution of a population of individuals in a new environment, by applying rules based on natural processes, such as DNA mutation, chromosomes crossover, natural selection, etc. Generation after generation, the DNA mixes and the best suited, or best adapted, genes cumulate in some individuals (Beasley and Chu 1996).

Applications of GAs are diversified and can handle problems of various types (Joines et al. 1996), even with very complex cost surfaces (Haupt and Haupt 2004). The objective function to optimize (often named fitness function in this context) can be of different types, but GAs must be adapted in order to perform optimally.

Two conditions guarantee in theory the convergence to the global optimum (Zitzler et al. 2004): (1) Parameters mutations that can allow to explore the entire parameter space, thereby ensures that any value can be achieved with a non-zero probability. (2) A rule of elitism ensuring that an optimal solution cannot be lost or damaged.

A key element of the parametrization of GAs is finding the right balance between exploration and exploitation (Bäck 1992a; Smith and Fogarty 1997). Exploration is characterized by a relatively high probability to assess the regions of the parameter space that have not yet been visited. This probability must be sufficiently large at the beginning of the optimization, so that the algorithm is capable of identifying the region where the global optimum is likely located. Exploitation is characterized by a local search in an area of interest, and generally makes small movements. The latter is interesting to refine the results at the end of the optimization.

b. Structure and operators

The GAs optimize a population of N individuals (parameter sets). Each individual contains a chromosome (set of parameters of the AM in this case). Genes are the individual parameters constituting the chromosome. They can be either categorical (e.g. geopotential, or temperature), discrete (e.g. number of analogues to select, from 1 to 40), or continuous. The floating-point representation (instead of the canonical binary, see Goldberg 1989, 1990; Holland 1992) of the genes was considered, as it was found more suited in multiple applications (Michalewicz 1996; Herrera et al. 1998; Haupt and Haupt 2004; Bäck and Schütz 1996; Gaffney et al. 2010).

There are numerous implementation variants of GAs, often optimal for a given problem (Hart and Belew 1991; Schraudolph and Belew 1992). The divergences are in the operators implementation, through significantly different algorithms, which has an important effect on the results (Gaffney et al. 2010). Here, operators are defined as the mechanisms that modify the values of the genes to try bringing individuals (or chromosomes) closer to an optimum of the fitness function. The structure of the method (Figure 1) resulting from the work of Holland (1992) is common to most applications (Bäck and Schwefel 1993), and consists in the following steps:

1. A population of N individuals (parameter sets of the AM) is randomly generated, which constitutes the initial population.
2. The fitness (performance score or objective function) of every individual is assessed.
3. A natural selection is applied, after which only the best individuals remain, which constitutes the intermediate generation (IG).
4. From this IG, couples are formed according to given rules.
5. These couples proceed to reproduction, or chromosome crossover, in order to mix their genes (parameters) according to the selected operator version. New children are generated in order to refill the IG back to N individuals.
6. Parents and children are then subject to mutation, where some genes get affected, meaning some parameter values are randomly changed.
7. The new formed generation is then re-assessed.
8. According to the ending criteria, the optimization ends or starts again for another iteration.

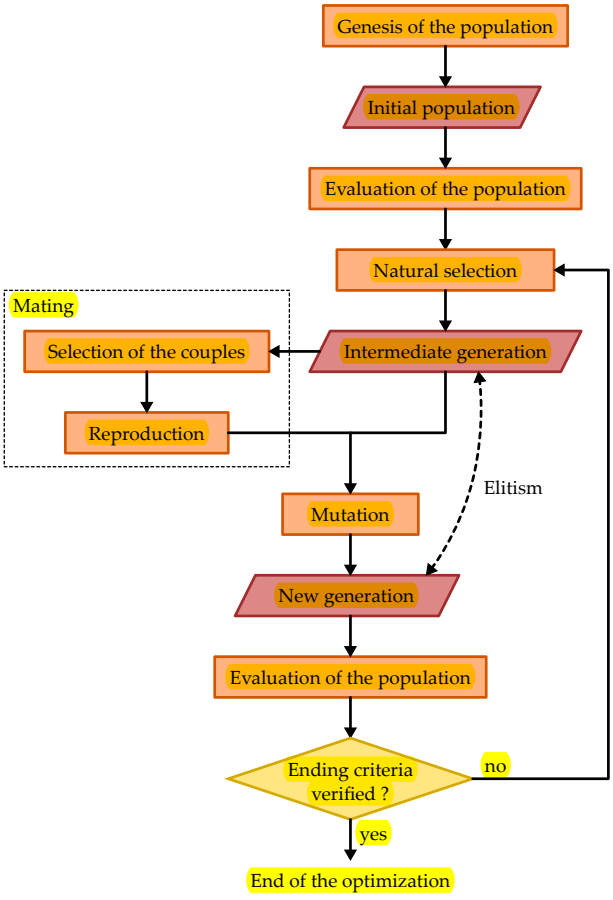

FIG. 1. Genetic Algorithms operational flowchart

All considered operators and their options, applied to floating-point coding, are described in the following sections. Many other operators exist, but only the ones evaluated are presented.

1) GENESIS OF THE POPULATION

The first step of the optimization is to generate an initial population. A population is a set of N individuals (each of which represents, in this application, a parameter set of the AM and thus a point in the space of potential solutions) that are going to evolve. A generation is the population considered at a given time.

A random initialization based on a uniform sampling is the most current version. The size N of the population is often a compromise between the computation time and the quality of the solution. N must allow sufficient sampling of the solutions field (Beasley and Chu 1996), and should thus vary as a function of chromosome size (ie the number of genes or parameters to be optimized).

2) NATURAL SELECTION

Natural selection is performed on the basis of the values of the objective function. The selection allows to only keep a certain part of the population, usually half ($N/2$), which can access the IG (with N_{IG} members). If N_{IG} is too high, the reproduction rate is too low, whereas if it is too small, the strong traits of individuals will not have the ability to accumulate in the same chromosome (Haupt and Haupt 2004). Several techniques exist, such as:

- N_{IG}-elitism (Michalewicz 1996): the population is ranked according to the value of the objective function and only the better half is preserved.

- Tournament selection (Michalewicz 1996; Zitzler et al. 2004): two individuals are randomly selected and fight. The one with the highest performance score is chosen, but with a certain probability, in order to reduce the selection pressure. This procedure is repeated until the IG is full. Individuals can be selected several times, and thus be represented several times in the IG.

3) SELECTION OF THE COUPLES

Individuals of the IG can reproduce. It begins with the selection of pairs (the parents). The techniques implemented in this work are the following:

- Rank pairing: individuals are gathered in pairs according to their rank (based on the performance scores). Consecutive ranks are put together (odd rows are associated with even rows). This approach is easy to achieve, but does not look like a natural process.

- Random pairing: two individuals are randomly selected to form a couple, according to a uniform law.

- Roulette wheel weighting: this technique refers to gambling, but with a biased roulette. Each individual is associated with a sector of the wheel with a certain opening angle, which is its probability of selection. The probability assigned to the individuals is proportional to their fitness (objective function), so that the most adapted individuals have the greatest probability of reproduction. There are two techniques for weighting the individuals of the IG:

  Roulette wheel weighting on rank: the probability of each individual depends on its rank n:

$$p_n = \frac{N_{IG} - n + 1}{\sum_{n=1}^{N_{IG}} n} \tag{1}$$

Roulette wheel weighting on fitness: the selection probability is calculated based on the value of the objective function. The probability p_n of each individual is calculated by the equation 2:

$$p_n = \frac{score_n - score_{N_{IG}}}{\sum_{n=1}^{N_{IG}} (score_n - score_{N_{IG}})} \quad (2)$$

In this application, the last individual (N_{IG}) has zero probability of being selected.

- Tournament selection: This operator is similar to the one used in natural selection, but is applied here for the successive selection of each parent. To select a parent, a number of individuals (2 or 3) are randomly picked and the best is kept. This operation is performed twice, once for each partner and imitates the breeding competition in nature (Haupt and Haupt 2004).

4) CHROMOSOME CROSSOVER

Once the two parents are selected for breeding, they combine their chromosomes and produce two children, bringing the number of individuals in the population back to N (the parents also return back in the total population in order to complement the next generation). The combination of chromosomes is carried out using a crossover operator, thereby generating two offspring having characteristics derived from both parents. It allows a mixing of genes and a potential accumulation of positive mutations. The evaluated crossover operators are the following:

- Single-point crossover: within the chromosome, a crossover point is randomly chosen for the pair. The genes (AM parameters) located after that point are exchanged in between the two chromosomes.
- Two-point crossover: works like the single-point crossover, but there are two intersections defining the segments to be exchanged. This approach, which significantly extends the search space for the children, is considered more efficient than the previous (Beasley et al. 1993).
- Multiple-point crossover (De Jong 1975): a generalization of the previous, with a number of crossover points up to the number of genes.
- Uniform crossover (Syswerda 1989): for each gene of the chromosome, it is randomly chosen to exchange or not the values between the parents.
- Binary-like crossover (Haupt and Haupt 2004): chromosome crossover on a binary coding can generate new values for variables located at intersection points, since the crossovers are applied at the bit level, thus often within a gene. This is not the case for the floating-point representation, since the crossover is performed between the genes. To reproduce the behaviour present in the original algorithms, which introduces new information, Haupt and Haupt (2004) propose an operator that combines standard crossover with an interpolation approach. The genes located after a crossover point are exchanged, but the gene located at the intersection is modified as follows (equation 3):

$$\begin{cases} g_{o1,n} = g_{p1,n} - \beta(g_{p1,n} - g_{p2,n}) \\ g_{o2,n} = g_{p2,n} + \beta(g_{p1,n} - g_{p2,n}) \end{cases} \quad (3)$$

where $g_{o1,n}$ and $g_{o2,n}$ are the n-th gene of the two new offspring, and $g_{p1,n}$ and $g_{p2,n}$ are those of the two parents. β is a random value between 0 and 1.

- Blending method (Radcliffe 1991): in this approach, instead of exchanging the genes in between the chromosomes after one or multiple crossover points, these are combined by linear combination (equation 4). The genes of the parents are blended together using a random value (β) that can be unique for the whole chromosome, or that can change for every gene. The genes of the offspring are bounded by the genes of the parents, no value can be out of their range.

$$\begin{cases} g_{o1,n} = \beta g_{p1,n} + (1-\beta) g_{p2,n} \\ g_{o2,n} = (1-\beta) g_{p1,n} + \beta g_{p2,n} \end{cases} \quad (4)$$

- Linear crossover (Wright 1991): in order to allow the genes to take values outside the interval defined by the parents, a method of extrapolation is necessary. Linear crossover introduces such an approach, and produces three children from two parents, following equation 5. Less couples are required in order to fill up the population.

$$\begin{cases} g_{o1,n} = 0.5 g_{p1,n} + 0.5 g_{p2,n} \\ g_{o2,n} = 1.5 g_{p1,n} - 0.5 g_{p2,n} \\ g_{e3,n} = -0.5 g_{p1,n} + 1.5 g_{p2,n} \end{cases} \quad (5)$$

- Heuristic crossover (Michalewicz 1996): it is a variation of the latter methods that relies on the following equation:

$$\begin{cases} g_{o1,n} = \beta(g_{p1,n} - g_{p2,n}) + g_{p1,n} \\ g_{o2,n} = \beta(g_{p2,n} - g_{p1,n}) + g_{p2,n} \end{cases} \quad (6)$$

- Linear interpolation: unlike previous techniques, this technique does not rely on crossover points, but on a linear interpolation on every gene of the couple (equation 7).

$$\begin{cases} c_{o1} = c_{p1} - \beta(c_{p1} - c_{p2}) \\ c_{o2} = c_{p2} + \beta(c_{p1} - c_{p2}) \end{cases} \quad (7)$$

where c_{o1} and c_{o2} are the full chromosomes of the offspring, and c_{p1} an c_{p2} are the ones of the parents. As before, β is a random value between 0 and 1, and is here the same for every gene.

- Free interpolation: this technique performs interpolation on each gene, like the previous one; but in this case, the weighting factor changes for each gene:

$$\begin{cases} c_{o1} = c_{p1} - [\beta_1(g_{p1,1} - g_{p2,1}), \beta_2(g_{p1,2} \\ \qquad - g_{p2,2}), ..., \beta_{Ng}(g_{p1,N_g} - g_{p2,N_g})] \\ c_{o2} = c_{p2} + [\beta_1(g_{p1,1} - g_{p2,1}), \beta_2(g_{p1,2} \\ \qquad - g_{p2,2}), ..., \beta_{Ng}(g_{p1,N_g} - g_{p2,N_g})] \end{cases} \quad (8)$$

where N_g is the number of genes, and β is here independent between the genes.

5) MUTATION

The combination of strong genes by the operator of chromosomes crossover is theoretically the most important operating mechanism in the conventional GAs (Holland 1992; Bäck and Schwefel 1993). However, many studies identify the mutation process as main operator, and crossovers as secondary (see Bäck 1992a, 1996; Bäck and Schütz 1996; Smith and Fogarty 1997; Deb and Beyer 1999; Costa et al. 2005, 2007).

The mutation operator is a direct modification of genes. In real coding, it is done by changing the gene values. Mutations add diversity to the population and prevent a freeze of the evolution, or a genetic drift to a local optimum. Thus, it makes the convergence to the global optimum theoretically possible (Beasley et al. 1993), as they allow exploring beyond the current region of the parameter space by bringing new characteristics that were not present in the original population (Haupt and Haupt 2004).

The evaluated and developed mutation operators are listed hereafter. They apply to genes made of continuous or discrete variables, but not categorical (eg meteorological variable or analogy criterion). In the latter case, the random choice of a new value is always based on a uniform distribution, without notion of distance in the parameters space.

- Uniform mutation: The mutation rate is constant and equal for every gene of each individual; they all have the same probability to mutate. When a gene is selected for mutation, a new random value is assigned, according to a uniform law.

- Variable uniform mutation (Fogarty 1989): a variable mutation rate over the generations was first suggested by Holland (1992) and evaluated by Fogarty (1989). In most applications, the mutation rate decreases with the generations, in a deterministic and global (for all individuals) manner (Bäck 1992b). Its optimum configuration depends on the size of the chromosomes, of the properties of the objective function, and of the population size (Bäck 1992b). This operator was implemented according to ~~equation 9~~,

$$p_{n,G} = p_{G_0} + \left(\frac{p_{G_0} - p_{G_{m,p}}}{G_{m,p}} \right) min\{G, G_{m,p}\} \quad (9)$$

where $p_{n,G}$ is the mutation rate (probability) of the gene n for generation number G, $G_{m,p}$ is the maximum number of generations during which the mutation rate varies. p_{G_0} is the initial mutation probability, and $p_{G_{m,p}}$ is the final one. p_{G_0}, $p_{G_{m,p}}$ and $G_{m,p}$ are the three controlling parameters of the operator. The evolution of the mutation rate is linear.

- Constant normal mutation: many applications use normal distributions to generate new values. The gene g that mutate becomes:

$$g' = N(g, \sigma^2) \quad (10)$$

where σ is the standard deviation of the distribution, which has to be estimated.

- Variable normal mutation (new): with the same logic as the variable uniform mutation, a mutation operator was tested using a normal distribution with a variable mutation rate and standard deviation. The mutation rate is calculated with equation 9. On the same principle, the standard deviation decreases linearly over the generations:

$$\sigma_{n,G} = \sigma_{G_0} + \left(\frac{\sigma_{G_0} - \sigma_{G_{m,\sigma}}}{G_{m,\sigma}} \right) min\{G, G_{m,\sigma}\} \quad (11)$$

where $\sigma_{n,G}$ is the standard deviation of gene n and generation number G, σ_{G_0} is the initial standard deviation, $\sigma_{G_{m,\sigma}}$ is the final standard deviation, $G_{m,\sigma}$ is the maximum number of generations during which the standard deviation varies. p_{G_0}, $p_{G_{m,p}}$, $G_{m,p}$, σ_{G_0}, $\sigma_{G_{m,\sigma}}$ and $G_{m,\sigma}$ are the six parameters of the method.

- Non-uniform mutation (Michalewicz 1996): two random numbers are picked based on a uniform law: r_1, which determines the direction of the change, and r_2, which determines its magnitude. The new value of the gene is given by the following equation, according to a predefined number of generations:

$$g_n^{'} = \begin{cases} g_n + (b_n - g_n)\, r_2 \left(1 - \dfrac{G}{G_m}\right)^2 & if\ r_1 < 0.5 \\ g_n - (g_n - a_n)\, r_2 \left(1 - \dfrac{G}{G_m}\right)^2 & if\ r_1 \geq 0.5 \end{cases} \quad (12)$$

where a_n is the is the lower bound of the n-th gene, b_n its upper bound, G the present generation, and G_m the maximum number of generations.

This operator was adapted for this application, which is not based on a predefined number of generations:

$$g_n^{'} = \begin{cases} g_n + (b_n - g_n)\, r_2 \varphi^2 & if\ r_1 < 0.5 \\ g_n - (g_n - a_n)\, r_2 \varphi^2 & if\ r_1 \geq 0.5 \end{cases} \quad (13)$$

with

$$\varphi = 1 - \min\left\{ \frac{G}{G_{m,r}}, 1 \right\} (1 - \omega) \quad (14)$$

where $G_{m,r}$ is the maximum number of generations during which the magnitude of the research varies, and ω is a threshold chosen by the user to maintain a minimum search radius when $G > G_{m,r}$. During the first generations, the exploration extent covers the entire parameter space. However, this area is reduced over generations, allowing exploitation of local solutions.

- Individual adaptive mutation rate (Bäck 1992a): based on the ideas of Evolution Strategies (see Rechenberg 1973; Schwefel 1981), Bäck (1992a) introduced a concept of self-adaptive GAs. The idea is to distribute control parameters within individuals themselves, which partially decentralize control of the evolution. It allows reducing the parametrization of GAs and introducing a notion of self-management. The first approach is the introduction of a mutation rate per individual, that mutates itself under its own probability (Bäck 1992a). Then, the eventual new rate is used to mutate the genes of the individual. Thus, as this rate decreases, it will have less probability of being itself mutated. Mutations are performed according to a constant uniform distribution. The initial mutation rates are randomly chosen (Bäck 1992a) and the method has no parameter. Other approaches exist to introduce a self-adaptation (see Smith and Fogarty 1997; Deb and Beyer 1999, 2001).

- Individual adaptive search radius (new): based on the ideas of the non-uniform mutation, a search radius was introduced in the approach of individual adaptive mutation rates. This search radius r_a, bounded between 0 and 1 (relatively to the parameters ranges), is also adaptive and behaves similarly to the adaptive mutation rates. In order to separate its evolution from the one of the mutation rate, its own value is considered initially as a self-mutation rate to eventually mutate before being used as a normalized search radius. The value of a mutated gene is given by the following equation, which is a simplification of the non-uniform mutation:

$$g_n^{'} = \begin{cases} g_n + (b_n - g_n)\, r_2 r_a & if\ r_1 < 0.5 \\ g_n - (g_n - a_n)\, r_2 r_a & if\ r_1 \geq 0.5 \end{cases} \quad (15)$$

where r_1 and r_2 are randomly selected, in the same way as for the non-uniform mutation. No external parameter is therefore necessary.

- Chromosome of adaptive mutation rate (or *n adaptative mutation rate*, Bäck 1992a): analogously to the individual adaptive mutation rate, this approach leaves the control of the evolution rate to the individuals themselves. The difference here is that each gene has a specific mutation rate. The main advantage is that the tuning of the mutation can be much more precise (Smith and Fogarty 1997). A second chromosome containing the mutation rate for each gene of the first chromosome was therefore considered. The operations of mutation and self-mutation are similar to the case of the individual adaptive mutation rate, but in a distributed way, within the chromosome. Another difference is that the same crossover operations are applied as those applied to the first chromosome, and this for the same crossing points. Thus, during an exchange of genes, children also inherit the mutation rates specific for each of these genes.

- Chromosome of adaptive search radius (new): this operator combines the operations of the chromosome of adaptive mutation rate to the adaptive search radius approach. Similarly, an individual has 3 chromosomes: the first containing the values to be optimized, the second the distributed mutation rate, and the last one, the distributed search radius. Again, no external parameters are required.

- Multi-scale mutation (new): finally, another approach was developed that is also based on the search radius concept. However, the latter is not decreasing with time. Methods based on a reduction of the mutation rate or radius simulate a transition from the exploration phase to the exploitation one. The idea was to test an approach that combines both exploration and exploitation during the whole optimization. Thus, the search radius r_a of equation 15 was considered as a random value for each individual, but restricted to 4 equiprobable values: 1, 0.5, 0.1, 0.02, which range from full exploration to fine exploitation. The only external parameter is the mutation rate which is fixed.

6) ELITISM

A process of elitism was introduced on the natural selection as well as on mutations. This ensures the survival of the best individual so that a better solution is never lost. After the natural selection operator, if the previously best individual has not been selected, it is copied to the IG instead of an individual randomly picked. After mutation, if the previously best individual has mutated and if its new

version has a lower performance score than the original, the latter is also reinserted in the IG instead of an individual randomly chosen.

7) ENDING THE OPTIMIZATION

The convergence check determines whether the solution is acceptable and if the algorithm may stop. The stopping criteria are not often well documented in GAs case studies. The optimization is here stopped if the best individual does not change for x generations. This value should not be too low to allow the algorithm to escape from a local optima. In addition, the rate of improvement decreases with the progression of the optimization. It is thus common that the best individual does not evolve over several generations when getting closer to the global solution. A value of $x = 20$ generations was chosen.

c. Implementation and constraints

Some constraints need to be taken into account. For example, when a crossover or a mutation operation results in a parameter value standing out of the authorized bounds, it has to be brought back within the limits. Moreover, the parameters are of different nature: some are continuous, some are discrete, and finally, some are categorical, i.e. independent elements in an array, such as the selection of the meteorological variable. New values resulting from the optimizer need to respect the type of data it represents.

Other constraints exist in between the parameters, such as the temporal window of the moisture index (see Horton et al. 2016a) that has to be consistent in between the relative humidity and the precipitable water.

GAs are very computationally intensive because they require many evaluations of the objective function. These assessments are very long in this application, as they require calculating and assessing a forecast for every day of the calibration period, thus over several decades. In order to reduce the computation time, recalculating the performance score of an individual who has previously been evaluated and that has not changed was avoided. Thus, the score of each individual living in the selection was kept until it mutates.

As the assessment (calculation of the objective function) of each member of the population of a generation is completely independent, it was performed in parallel on different processors of a computer, as suggested by Alliot and Durand (2005). In order to perform optimizations for multiple time series, the use of a cluster is a necessity, which our code allows.

3. Assessment process and results

The GAs parametrization, i.e. the a priori choices such as the mutation rate, population size, natural selection options, etc, appears difficult given the high number of existing variants, each developed for a specific problem (Haupt and Haupt 2004; Costa et al. 2007). The parametrization depends on the objective function, implementation variants, the range of the parameters to be optimized, and performance indicators. Thus, different studies suggest very different parameterizations.

De Jong (1975) and Grefenstette (1986) compared different implementations and parametrizations of GAs on functions of varying complexity. They observed that a small population size improves the initial performance, while a large population improves latter performance. They also observed that the proportion of the population to keep for the IG is around 50% (45% to 60%).

Values of the mutation rate varies broadly between studies: from 0.001 (De Jong 1975) to 0.2 (Haupt and Haupt 2004). Bäck and Schütz (1996) showed that mutation rates higher than the usual ranges are more optimal at the beginning of the optimization, allowing further exploration. Varying mutation rates are certainly more optimal but more complex to implement (Bäck 1996; Bäck and Schütz 1996).

a. Comparison process

One of our goals being to make recommendations of parametrization in view of optimizing the AM, a systematic procedure was adopted. The results are summarized hereafter (see Horton 2012, for the details). Concepts were taken from the factorial design approach (see eg. Costa et al. 2005, 2007; Mariano et al. 2010), which is sometimes used for comparative analysis of different parametrizations of GAs. It allows isolating the effect of a parameter under different combinations of the other options. A procedure by stages was adopted, analyzing in details and in a systematic way every variants of the implemented operators, in combination with multiple other options and parameters in order to take into account eventual co-dependencies. The goal here is not to focus on the performance score obtained through optimization, neither the values of the new optimized parameters (covered in Horton et al. 2016b), but to explain how to use GAs to optimize AMs in an efficient way.

In order to evaluate a combination of operators/options, 10 optimizations per parametrization of GAs were processed. Such assessment is not possible on the whole archive length, and had to be performed on a reduced period. The performances were characterized by four indicators: (i) mean performance score: average of the final scores of the 10 optimizations, (ii) convergence: the number of optimizations that converged to a supposed global optimum, (iii) number of generations: characterization of the convergence speed, and (iv) number of evaluations of the objective function: characterization of the required calculation time.

b. Success of the approach

After a first overview of the results, GAs have quicky proved successful at optimizing the AM. The performance score of the sequential approach has been quickly exceeded without adding any new parameter to the method (Figure 2). Even GAs parameterizations that will be considered later as inadequate (Figure 3) did significantly better that the sequential approach. The amplitude of the improvement is not the main outcome here. The most important point is that GAs proved successful at optimizing AMs automatically, globally, and objectively.

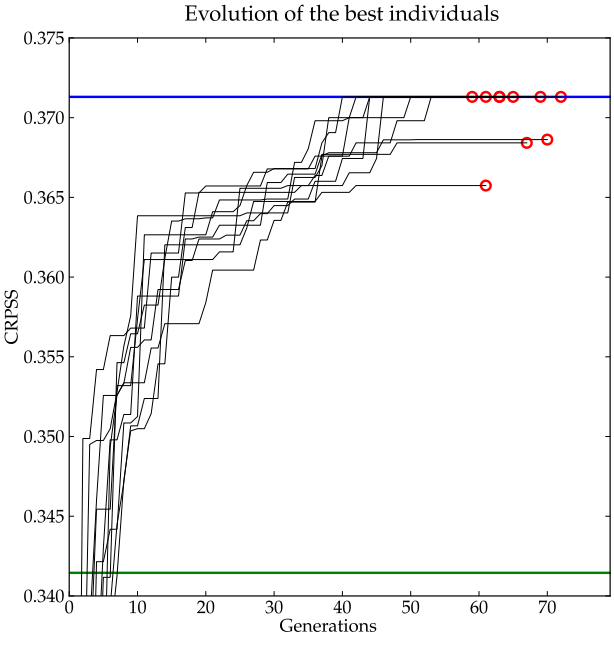

FIG. 2. Evolution of the score of the best individuals over generations for the 10 optimizations processed for a given parametrization. The green line (bottom) represents the score of the sequential approach and the blue one (top), the supposed global optimum.

c. Results of the comparison

The results illustrate the effect of an operator when its contribution is isolated from the other operators. It means that we analyze the effect of a given operator for equivalent conditions (same settings of other operators). Multiple combinations with other operators are assessed. This contribution is then summarized as a percentage of gain/loss regarding the mean of all variants, for equivalent external conditions. For example, to evaluate the performance of the uniform crossover operator, its performance is compared to the average of all crossover operators while retaining the same population size, the same mutation operators, natural selection, and selection of couples.

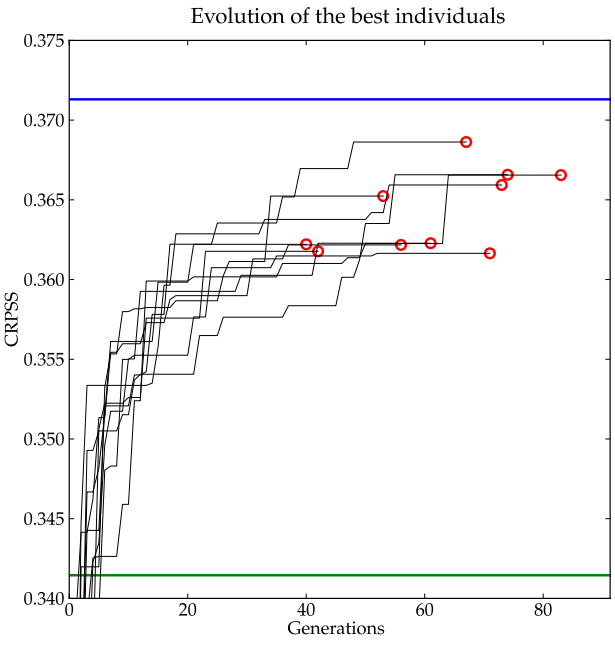

FIG. 3. Same as Figure 2, but for a GAs parametrization considered as less relevant.

From the very beginning of the assessments, the importance of the mutation operator was obvious (see Horton 2012, for the details), and its leading influence on the optimization performance was evident. Its role is analyzed later on.

1) BREEDING OPERATORS

Every combination of 6 options for the couples selection (Table 1) and 21 for the chromosome crossover operators (Table 2) were evaluated, along with variants of the other operators. This resulted in 1,008 combinations, requiring 10,080 optimizations.

TABLE 1. Assessed operator for couples selection.

	Couples selection operators
A	Rank pairing
B	Random pairing
C	Roulette wheel weighting on rank
D	Roulette wheel weighting on fitness
E	Tournament selection (3 candidates)
F	Tournament selection (4 candidates)

The performance of the couples selection operator are relatively close (Figure 4). Overall, the tournament selection with 3 candidates is slightly superior to others, along

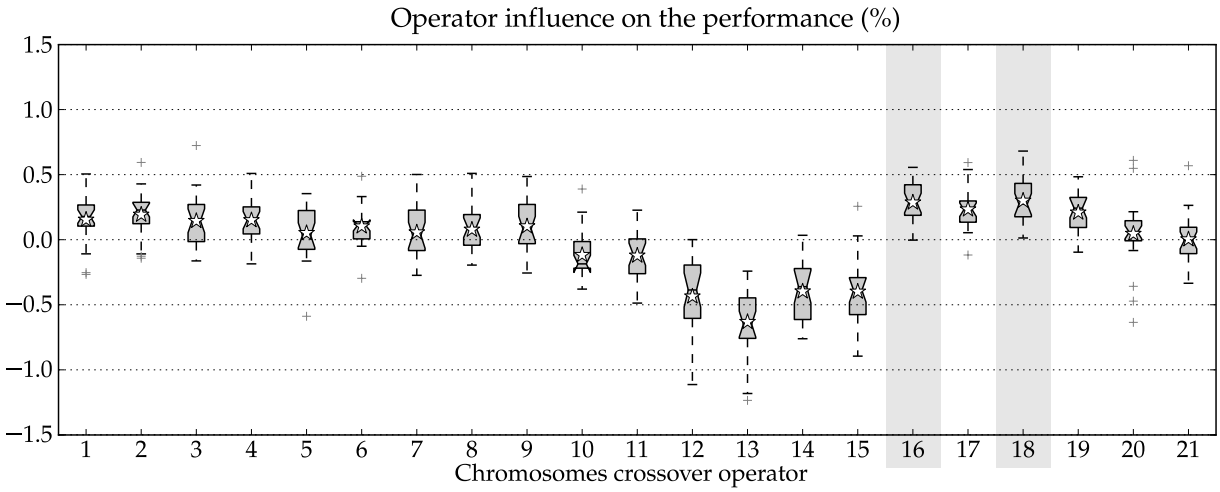

FIG. 5. Influence of the chromosome crossover operators (Table 2) on the optimization performance. Same conventions as Figure 4.

TABLE 2. Assessed operators for chromosome crossover.

	Chromosome crossover operators
1	Single-point crossover
2	Two-point crossover
3	Multiple-point crossover (3 points)
4	Multiple-point crossover (5 points)
5	Uniform crossover
6	Blending method (2 points, unshared β)
7	Blending method (4 points, unshared β)
8	Blending method (2 points, shared β)
9	Blending method (4 points, shared β)
10	Linear crossover (2 points)
11	Linear crossover (4 points)
12	Heuristic crossover (2 points, unshared β)
13	Heuristic crossover (4 points, unshared β)
14	Heuristic crossover (2 points, shared β)
15	Heuristic crossover (4 points, shared β)
16	Binary-like crossover (2 points, unshared β)
17	Binary-like crossover (4 points, unshared β)
18	Binary-like crossover (2 points, shared β)
19	Binary-like crossover (4 points, shared β)
20	Linear interpolation
21	Free interpolation

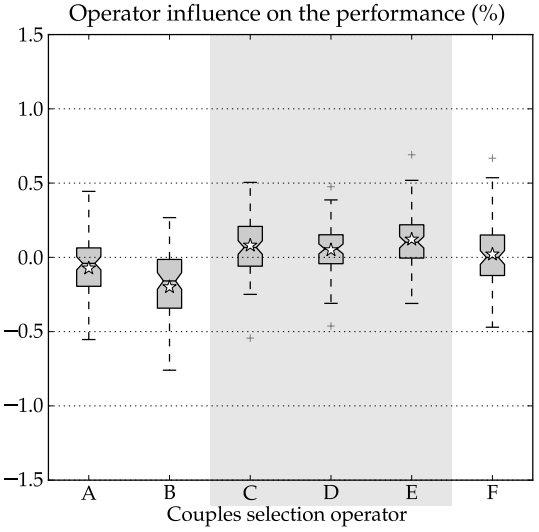

FIG. 4. Influence of the couples selection operators (Table 1) on the optimization performance. The box extends from the lower to upper quartile values of the data, with a line at the median. The whiskers extend from the box to 1.5 times the interquartile range. Flier points are those past the end of the whiskers. The star represents the median. The gray box highlights the best options.

with the roulette wheel weighting. This last one is however a bit less effective in terms of convergence and number of evaluations (not shown). The couples selection operator has not a significant role in this application.

Analysis of crossover operators (Figure 5) reveal some slightly superior options, some inappropriate, and many average. Binary-like crossover (especially with 2 points of intersection, whether β is shared or not) are significantly better than the others, especially in terms of convergence (not shown). Others operators can also be considered usable.

2) MUTATION OPERATOR

Having identified the leading role of the mutation operator, the next sensitivity analysis focused on it. Each of the 10 different implementations (see section 5) was tested, with different parameters for those who require some (Ta-

ble 3), bringing the number of variations up to 109. Some optimizations without any mutation were also performed as a reference. Along with variants of the other operators (see Horton 2012, for the details), this resulted in 660 combinations (so 6,600 optimizations).

TABLE 3. Assessed mutation operators with the number of variants considered (combination of parameters).

	Mutation operator	Variants
1	Uniform mutation	3
2	Variable uniform mutation	27
3	Constant normal mutation	9
4	Variable normal mutation	36
5	Non-uniform mutation	27
6	Individual adaptive mutation rate	1
7	Individual adaptive search radius	1
8	Chromosome of adaptive mutation rate	1
9	Chromosome of adaptive search radius	1
10	Mutli-scale mutation	3
11	No mutation	1

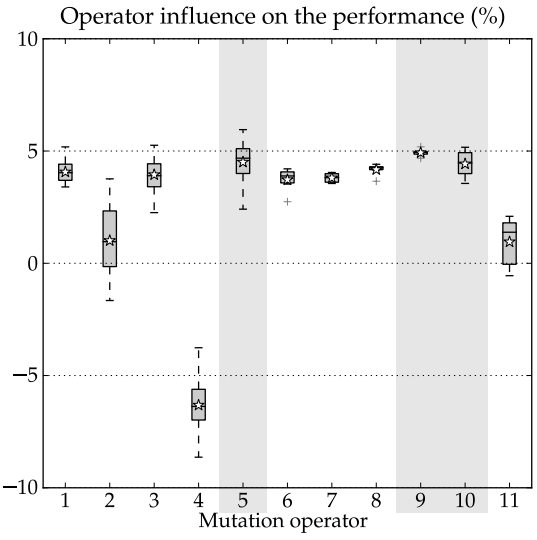

FIG. 6. Influence of the mutation operators (Table 3) on the optimization performance. Same conventions as Figure 4.

Figure 6 show the results of this analysis and illustrates the important role of the mutation on the performance of the optimizations. Those without mutation (last box on the Figure) are inferior to most mutation operators, and the scale of the influence of this operator is significantly more important than those for the other options. The details of the analysis (see Horton 2012) show that the other reproduction operators seem of secondary importance. This observation is in line with the work of Bäck (1996), who argues for the importance of mutation over reproduction, as well as other authors (see section 2.5).

The mutation operators based on a variable normal or variable uniform laws work very poorly and are difficult to configure. Many operators present more or less the same performance scores and require a variable amount of assessments. The convergence analysis (see Horton 2012) allows to highlight three best operators: non-uniform mutation, chromosome of adaptive search radius, and multi-scale mutation. Thus, different optimizations were further performed using variants of these 3 operators (Table 4).

TABLE 4. Further assessments of mutation operators.

	Mutation operator	p_{mut}	G_{max}	ω
1	Non-uniform mutation	0.01	50	0.1
2	Non-uniform mutation	0.05	50	0.1
3	Non-uniform mutation	0.1	50	0.1
4	Non-uniform mutation	0.2	50	0.1
5	Non-uniform mutation	0.4	50	0.1
6	Non-uniform mutation	0.01	100	0.1
7	Non-uniform mutation	0.05	100	0.1
8	Non-uniform mutation	0.1	100	0.1
9	Non-uniform mutation	0.2	100	0.1
10	Non-uniform mutation	0.4	100	0.1
11	Mutli-scale mutation	0.01		
12	Mutli-scale mutation	0.05		
13	Mutli-scale mutation	0.1		
14	Mutli-scale mutation	0.2		
15	Mutli-scale mutation	0.4		
16	Chromosome of adaptive search radius			

The first analysis was the optimization of the precipitation forecasting over a subcatchment (Binn-Simplon region) in the Swiss Alps (Figure 7). The optimizer could choose the 2 pressure levels of the atmospheric circulation analogy (method with a single level of analogy). The resulting performance score (CRPSS = skill score of the CRPS, see Brown 1974) is obviously superior to the one obtained by the sequential calibration (red line on Figure 7). For most options, it is also slightly better than the results from the optimization without selection of the pressure levels (blue line). A clear breakthrough of the performances was not expected, as the former selection of pressure levels results already from intensive comparative work (Bontron 2004). This application however demonstrates that, when correctly parametrized, GAs can automatically and successfully choose the pressure levels. However, some less relevant optimizations do not match the previous results. Through different applications, the automatic selection of the pressure level was shown to significantly increase the difficulty for GAs to converge to a unique solution, ideally the global optimum. A difficulty is that the pressure levels are considered within the optimization without continuity between values, and thus the

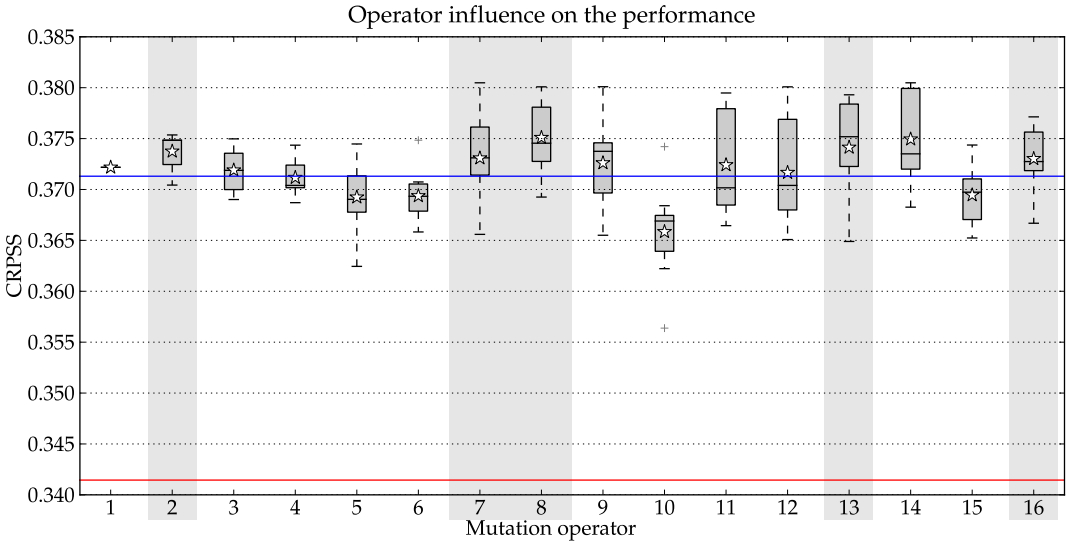

FIG. 7. Influence of the mutation operators (Table 4) on the optimization performance, leaving the optimizer choose the pressure level of the atmospheric circulation analogy (single level of analogy). The predictand is precipitation over a subcatchment in the Swiss Alps (Binn-Simplon region). The red line represents the score of the sequential calibration and the blue line, the score of the optimization without automatic selection of the pressure levels. Same conventions as Figure 4.

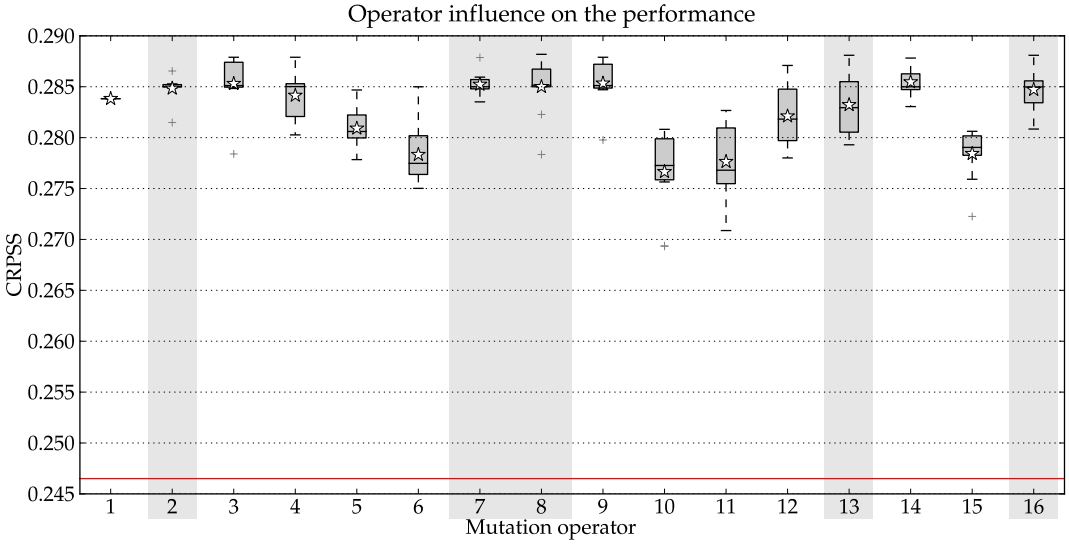

FIG. 8. Same as Figure 7, but for another region in the Swiss Alps (Bottom Rhone valley), with different atmospheric influences.

approaches relying on distance in the parameters space, such as the search radius, cannot fully exploit the properties that make them efficient. However, even though the results show a certain variability, most of them present very good performance scores, despite different parameters of the AM.

Then, the same optimization was performed, but for another region, sensitive to other meteorological influences (Figure 8), in order to assess eventual dependencies of the operator with the predictand. Even though differences can be observed with Figure 7, it is globally the same options that perform better.

Next, a second level of analogy was proposed (Figure 9) based on moisture variables (see Bontron 2004; Horton et al. 2016a). GAs had to optimize both levels of analogy simultaneously. Once again, despite the difficulty to do so, the results were better than the sequential calibration (red line on Figure 9). And finally, a preselection on air

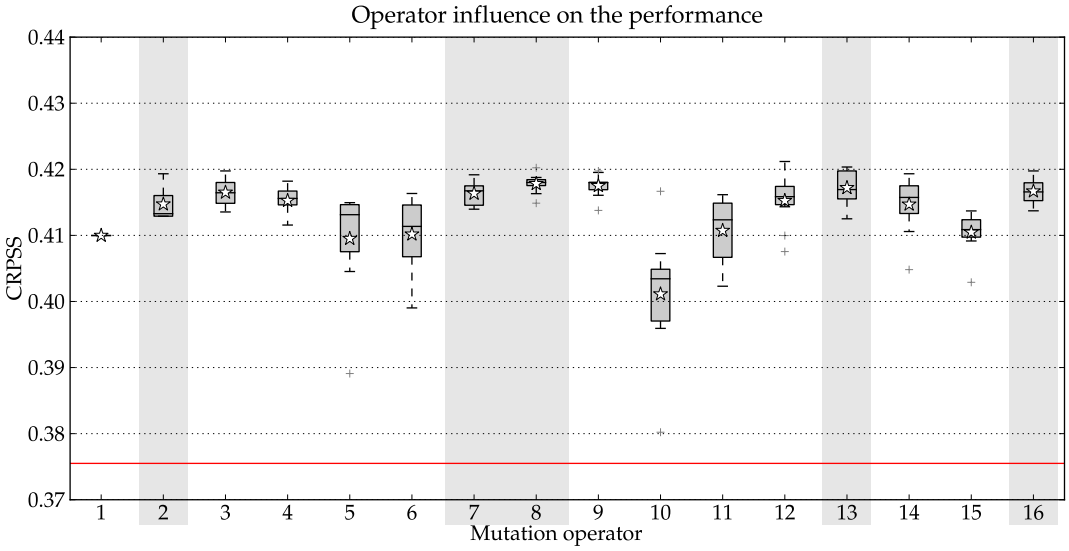

FIG. 9. Same as Figure 7, but with a second level of analogy on moisture variables.

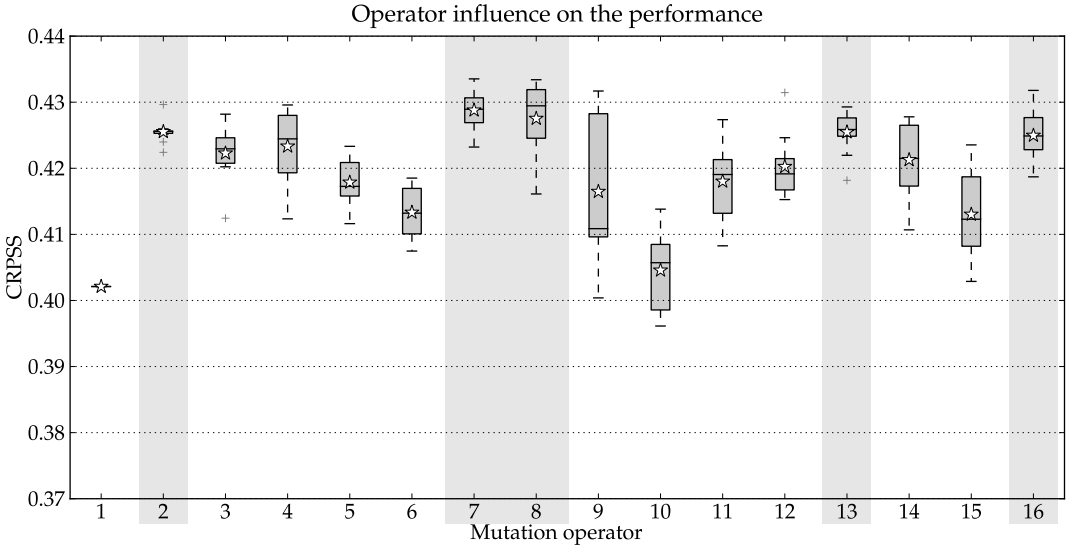

FIG. 10. Same as Figure 9, but with a preselection on air temperature rather than a fixed calendar window.

temperature was added instead of the fixed calendar window, as proposed by Ben Daoud et al. (2015). The results show generally higher scores (Figure 10), demonstrating the success of the optimizer to take advantage of this new degree of freedom, and its capacity to handle optimization of 3 analogy levels jointly. Again, the most relevant options are generally the same.

After analysis of the most relevant mutation operators, the following advice can be raised (detailed parametrization are provided in section 4):

- *Non-uniform mutation* (Michalewicz 1996): this operator is good in terms of convergence, mainly when the number of parameters to optimize is rather low. The number of required evaluations, however, can be quite substantial. The main disadvantage of the non-uniform mutation is the number of parameters it requires, that are difficult to estimate a priori. The mutation rate was found to be more important than the others. The difficulty is that its optimal value may be case-related.

- *Chromosome of adaptive search radius* (new): unlike the previous one, this new operator is very robust, as it requires no option and is auto-adapting. It is interesting to notice that the insertion of an extra chromosome representing the search radius gives better performance than other self-adaptive operators (such as, for example, the chromosome of adaptive mutation rate). If one had to choose a single option for the mutation operator, we would recommend this one, as it was proven effective and needs no parameter.

- *Multi-scale mutation* (new): finally, the multi-scale mutation, which also performs pretty well, requires one parameter, the mutation rate. However, it can also be difficult to estimate a correct value a priori.

In this application, the mutation operator has a leading effect and should be chosen with care. It may be wise to perform multiple optimizations and to consider these three operators in parallel in order to obtain results from options that are sometimes either more efficient or more robust. It is interesting to note that the three best techniques incorporate a notion of search distance. It is likely that this notion is the key to these algorithms, for this application, and allows them to initially explore the parameter domain, and then to converge. The search radius in fact directly represents the notion of transition between exploration and exploitation, in our opinion more than a possible evolution of mutation rates.

3) OTHER OPTIONS

The analysis of the natural selection operator (Figure 11) reveals a slight preference for the ratio-elitism compared to the tournament selection, but not so significant. This operator, or at least the two assessed versions, do not appear to significantly influence the optimization performances.

The size of the population (N), i.e. the number of sets of AM parameters considered, has an effect on the performance of the optimization (Figure 12). A bigger population leads to better results, but also to significantly longer optimizations. Indeed, the required number of evaluation, and thus the required time, is approximately proportional to the population size. The optimal size seems to depend on the complexity of the AM to optimize: a more complex AM (ie. with more degrees of freedom) requires a bigger population size. A rule of thumb based on a limited number of case studies (not shown here) is provided hereafter:

- $N \approx 100$ for very simple implementations of the AM (1 level of analogy with 2 pressure levels),

- $N \approx 200$ for a slightly more complex AM (1 level of analogy with 4 pressure levels, or 2 level of analogy with less pressure levels),

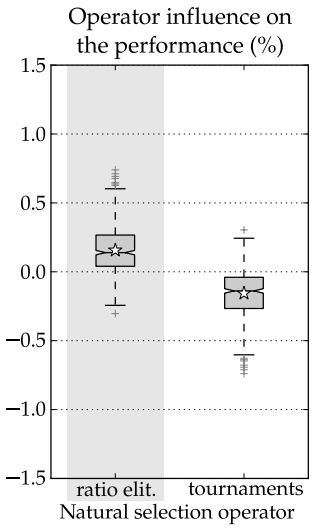

FIG. 11. Influence of the natural selection operators on the optimization performance. Same conventions as Figure 4.

- $N \approx 500$ for significantly more complex AMs (2-3 levels of analogy with 4 pressure levels for the atmospheric circulation, and 2 to 4 levels for the moisture analogy).

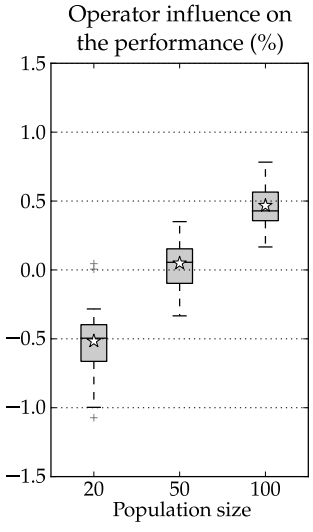

FIG. 12. Influence of the population size on the optimization performance. Same conventions as Figure 4.

The influence of the size of the IG (proportion of the total population) selected for mating was also assessed (Figure 13). It does not appear that this parameter is critical to the quality of the optimizations, provided it is not too big. A value of 50 % seems a wise choice.

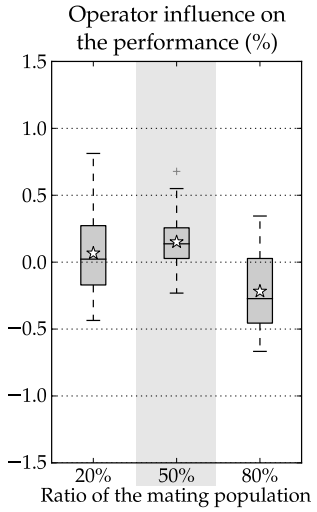

FIG. 13. Influence of the intermediate population (IG) ratio on the optimization performance. Same conventions as Figure 4.

4. Recommended parametrization of GAs

Optimizations by GAs for AMs of varying complexities were performed with a large number of combinations of operators in order to make recommendations for optimizing the AM. The conclusions are:

- The population size should be in accordance with the complexity of the AM to optimize: from 100 for the simple ones, up to 500 for the most complex AMs.
- The value of the ratio for the IG is not so important, and value of 50% seems quite appropriate.
- Ratio-elitism is slightly better than tournaments for the natural selection operator, but it is not decisive.
- The performance of the operators for the couples selection perform relatively similarly. The roulette wheel weighting and the tournament selection are more efficient in terms of convergence and required number of evaluations.
- Most crossover operators have relatively similar performance. Binary-like crossover with two points of intersection are better than others, especially for convergence.
- Mutation has a clearly dominant influence. Three mutation operators stand out: the non-uniform mutation, the multi-scale mutation, and the chromosome of adaptive search radius. The latter is the most robust as it has no controlling parameter.

The optimization does not systematically converge to the global optimum (but still often nearby), which is why it is recommended to do several optimizations in parallel in order to compare the results, analyze the convergence, and keep the best. It may be wise to consider the three mutation operators in parallel. In order to be confident in the optimized AMs, we propose using a set of the following mutation operators:

- 1x non-uniform, $p_{mut} = 0.05$, $G_m = 50$, $\omega = 0.1$
- 1x non-uniform, $p_{mut} = 0.05$, $G_m = 100$, $\omega = 0.1$
- 1x non-uniform, $p_{mut} = 0.1$, $G_m = 100$, $\omega = 0.1$
- 1x multi-scale, $p_{mut} = 0.1$
- 2x chromosome of adaptive search radius

5. Conclusions

In order to automatically optimize the AM and to get rid of the limitations of the usual sequential calibration, GAs were evaluated. Given the large number of existing operators and options, multiple variants were assessed systematically in order to identify which operators are important, and which variants work best for the AM. The mutation operator was identified as a key element for this application, and new variants that proved efficient were provided, such as the chromosome of adaptive search radius that is very robust (no control parameter). Recommendations were established for a relevant use of GAs for the optimization of AMs.

It is not excluded that another global optimization method or other operators of GAs may perform still better. However, the relevance of such an approach has now been proved as it results in parametrization of AMs that are relevant and fully automatically, globally and objectively established. A global optimization is the only way to take into account all the dependencies between parameters and levels of analogy.

The global optimization approach allows easily adapting the AM to new regions by potentially taking into account local meteorological influences, and has thus a great potential of use. Moreover, it allows exploring automatically datasets in order to extract the most relevant variables. It is thus possible to try assessing other predictands, such as the temperature, the limit of snowfall, the occurrence of hail, or wind, while leaving the algorithms select the best variables and the associated parameters.

Acknowledgments. Thanks to Hamid Hussain-Khan of the University of Lausanne for his help and availability, and for the intensive use of the cluster he is in charge of. Thanks to Renaud Marty for his fruitful collaboration over the years. Thanks to Dominique Bérod for his support and to Michel Bierlaire for his advices on optimization methods.

Thanks to the Swiss Federal Office for Environment (FOEV), the Roads and Water courses Service, Energy and Water Power Service of the Wallis Canton and the Water, Land and Sanitation Service of the Vaud Canton who

financed the MINERVE (Modélisation des Intempéries de Nature Extrême des Rivières Valaisannes et de leurs Effets) project which started this research. The fruitful collaboration with the Laboratoire d'Etude des Transferts en Hydrologie et Environnement of the Grenoble Institute of Technology (G-INP) was made possible thanks to the Herbette Foundation. NCEP reanalysis data provided by the NOAA/OAR/ESRL PSD, Boulder, Colorado, USA, from their Web site at http://www.esrl.noaa.gov/psd/. Precipitation time series provided by MeteoSwiss.

References

Alliot, J. M., and N. Durand, 2005: Algorithmes génétiques.

Bäck, T., 1992a: Self-adaptation in genetic algorithms. *Proceedings of the First European Conference on Artificial Life*, 263–271.

Bäck, T., 1992b: The Interaction of Mutation Rate, Selection, and Self-Adaptation Within a Genetic Algorithm. *Parallel Problem Solving from Nature*, **2**, 85–94.

Bäck, T., 1996: Evolution strategies: An alternative evolutionary algorithm. *Artificial Evolution*, 3–20, URL http://link.springer.com/chapter/10.1007/3-540-61108-8{_}27.

Bäck, T., and M. Schütz, 1996: Intelligent mutation rate control in canonical genetic algorithms. *Foundation of Intelligent Systems, 9th International Symposium*, 158–167, URL http://link.springer.com/chapter/10.1007/3-540-61286-6{_}141.

Bäck, T., and H.-P. Schwefel, 1993: An Overview of Evolutionary Algorithms for Parameter Optimization. *Evolutionary Computation*, **1 (1)**, 1–23, doi:10.1162/evco.1993.1.1.1, URL http://www.mitpressjournals.org/doi/abs/10.1162/evco.1993.1.1.1.

Barnett, T., and R. Preisendorfer, 1978: Multifield analog prediction of short-term climate fluctuations using a climate state vector. *Journal of the Atmospheric Sciences*, **35**, 1771–1787, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0469(1978)035{\%}3C1771:MAPOST{\%}3E2.0.CO;2.

Beasley, D., R. Martin, and D. Bull, 1993: An overview of genetic algorithms: Part 1. Fundamentals. *University computing*, **15 (2)**, 58–69, URL http://www.geocities.ws/francorbusetti/gabeasley1.pdf.

Beasley, J., and P. Chu, 1996: A genetic algorithm for the set covering problem. *European Journal of Operational Research*, **94 (2)**, 392–404, doi:10.1016/0377-2217(95)00159-X, URL http://linkinghub.elsevier.com/retrieve/pii/037722179500159X.

Ben Daoud, A., 2010: Améliorations et développements d'une méthode de prévision probabiliste des pluies par analogie. Ph.D. thesis, Université de Grenoble.

Ben Daoud, A., E. Sauquet, G. Bontron, C. Obled, and M. Lang, 2015: Daily quantitative precipitation forecasts based on the analogue method: improvements and application to a French large river basin. *Atmospheric Research*, **169**, 147–159, doi:10.1016/j.atmosres.2015.09.015, URL http://linkinghub.elsevier.com/retrieve/pii/S0169809515002951.

Bergen, R., and R. Harnack, 1982: Long-range temperature prediction using a simple analog approach. *Monthly Weather Review*, **110**, 1083–1099, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0493(1982)110{\%}3C1083:LRTPUA{\%}3E2.0.CO{\%}3B2.

Bliefernicht, J., 2010: Probability forecasts of daily areal precipitation for small river basins. Ph.D. thesis, Universität Stuttgart.

Bois, P., C. Obled, and J. Thalamy, 1981: Etude des liaisons entre champs de pression et températures sur l'Europe avec la durée d'insolation: Application à la possibilité de la prévision d'insolation. *C.R. du colloque Météorologie de l'énergie solaire, PIRDES-CNRS, Toulouse*, 145–177.

Bolognesi, R., 1993: Premiers développements d'un modèle hybride pour le diagnostic spatial des risques d'avalanches. *La Houille Blanche*, **8**, 551–553.

Bontron, G., 2004: Prévision quantitative des précipitations: Adaptation probabiliste par recherche d'analogues. Utilisation des Réanalyses NCEP/NCAR et application aux précipitations du Sud-Est de la France. Ph.D. thesis, Institut National Polytechnique de Grenoble, 262 p. pp.

Bontron, G., and C. Obled, 2005: L'adaptation probabiliste des prévisions météorologiques pour la prévision hydrologique. *La Houille Blanche*, **(1)**, 23–28, URL http://www.shf-lhb.org/index.php?option=com{_}article{\&}access=dkey{\&}dkey=10.1051/lhb:200501002{\&}lang=fr{\&}Itemid=0.

Brown, T., 1974: Admissible Scoring Systems for Continuous Distributions. URL http://eric.ed.gov/?id=ED135799, P–5235 pp.

Costa, C. B., E. a.C. Rivera, M. C. A. F. Rezende, M. R. W. Maciel, and R. M. Filho, 2007: Prior detection of genetic algorithm significant parameters: Coupling factorial design technique to genetic algorithm. *Chemical Engineering Science*, **62 (17)**, 4780–4801, doi:10.1016/j.ces.2007.03.042, URL http://linkinghub.elsevier.com/retrieve/pii/S0009250907003168.

Costa, C. B., M. R. W. Maciel, and R. M. Filho, 2005: Factorial design technique applied to genetic algorithm parameters in a batch cooling crystallization optimisation. *Computers & Chemical Engineering*, **29 (10)**, 2229–2241, doi:10.1016/j.compchemeng.2005.08.005, URL http://linkinghub.elsevier.com/retrieve/pii/S0098135405001936.

De Jong, K. A., 1975: Analysis of the behavior of a class of genetic adaptive systems. Ph.D. thesis, University of Michigan.

Deb, K., and H. Beyer, 1999: Self-adaptation in real-parameter genetic algorithms with simulated binary crossover. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), Orlanda, USA*, 172–179, URL http://repository.ias.ac.in/82737/.

Deb, K., and H. Beyer, 2001: Self-adaptive genetic algorithms with simulated binary crossover. *Evolutionary Computation*, **9 (2)**, 197–221, URL http://www.mitpressjournals.org/doi/abs/10.1162/106365601750190406.

Duband, D., 1970: Reconnaissance dynamique de la forme des situations météorologiques. Application à la prévision quantitative des précipitations. Ph.D. thesis, Thèse de 3ème cycle de la faculté des sciences de Paris.

Fogarty, T. C., 1989: Varying the probability of mutation in the genetic algorithm. *Proceedings of the third international conference on Genetic algorithms*, Morgan Kaufmann Publishers Inc., 104–109.

Fraedrich, K., C. C. Raible, and F. Sielmann, 2003: Analog Ensemble Forecasts of Tropical Cyclone Tracks in the Australian Region. *Weather and Forecasting*, **18 (1)**, 3–11, doi:10.1175/1520-0434(2003)018⟨0003:AEFOTC⟩2.0.CO;2, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0434{\%

}282003{\%}29018{\%}3C0003{\%}3AAEFOTC{\%}3E2.0.CO{\%}3B2.

Gaffney, J., C. Pearce, and D. Green, 2010: Binary versus real coding for genetic algorithms: A false dichotomy? *ANZIAM Journal*, **51**, 347–359, URL http://dev.journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/2776.

Goldberg, D., 1990: Real-coded genetic algorithms, virtual alphabets, and blocking. *Urbana*, **51**, 1–21, URL http://illigal.org/wp-content/uploads/illigal/pub/papers/IlliGALs/90001.pdf.

Goldberg, D. E., 1989: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc.

Gordon, N. D., 1987: Statistical very short-range forecasting via analogues. *European Space Agency Special Publication*, 307–312.

Grefenstette, J. J., 1986: Optimization of control parameters for genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics*, **16 (1)**, 122–128.

Guilbaud, S., 1997: Prévision quantitative des précipitations journalières par une méthode statistico-dynamique des recherche d'analogues - Application à des bassins du pourtour méditerranéen. Ph.D. thesis, Institut National Polytechnique de Grenoble, 386 p. pp.

Hart, W. E., and R. K. Belew, 1991: Optimizing an arbitrary function is hard for the genetic algorithm. *Proceedings of the Fourth International Conference on Genetic Algorithms*, 190–195.

Haupt, R. L., and S. E. Haupt, 2004: *Practical genetic algorithms*. John Wiley & Sons, 272 p. pp.

Herrera, F., M. Lozano, and J. Verdegay, 1998: Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis. *Artificial Intelligence Review*, **12 (4)**, 265–319, URL http://link.springer.com/article/10.1023/A:1006504901164.

Holland, J., 1992: Genetic algorithms. *Scientific American*, **267 (1)**, 66–72.

Horton, P., 2012: Améliorations et optimisation globale de la méthode des analogues pour la prévision statistique des précipitations. Développement d'un outil de prévision et application opérationnelle au bassin du Rhône à l'amont du Léman. Thèse de doctorat, Université de Lausanne, 320 p. pp.

Horton, P., M. Jaboyedoff, R. Metzger, C. Obled, and R. Marty, 2012: Spatial relationship between the atmospheric circulation and the precipitation measured in the western Swiss Alps by means of the analogue method. *Natural Hazards and Earth System Sciences*, **12**, 777–784, doi:10.5194/nhess-12-777-2012.

Horton, P., M. Jaboyedoff, and C. Obled, 2016a: AtmoSwing (v1.4): Analog Technique model for Statistical weather forecastING. *Geoscientific Model Development*, **submitted**.

Horton, P., M. Jaboyedoff, and C. Obled, 2016b: Using Genetic Algorithms to Optimize the Analogue Method for Precipitation forecasting in the Swiss Alps. *Journal of Hydrology*, **submitted**.

Joines, J., C. Culbreth, and R. King, 1996: Manufacturing cell design: an integer programming model employing genetic algorithms. *IIE transactions*, **28 (1)**, 69–85, URL http://www.tandfonline.com/doi/abs/10.1080/07408179608966253.

Keenan, T. D., and F. Woodcock, 1981: Objective Tropical Cyclone Movement Forecasts Using Synoptic and Track Analogue Information. Tech. rep., Bureau of Meteorology.

Kruizinga, S., and A. Murphy, 1983: Use of an analogue procedure to formulate objective probabilistic temperature forecasts in the Netherlands. *Monthly Weather Review*, **111 (11)**, 2244–2254, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0493(1983)111{\%}3C2244{\%}3AUOAAPT{\%}3E2.0.CO{\%}3B2.

Livezey, R. E., and A. G. Barnston, 1988: An operational multifield analog/antianalog prediction system for United States seasonal temperatures. 1. system design and winter experiments. *Journal of Geophysical Research*, **93 (D9)**, 10 953–10 974.

Lorenz, E., 1956: Empirical orthogonal functions and statistical weather prediction. Tech. rep., Massachusetts Institute of Technology, Department of Meteorology, Massachusetts Institute of Technology, Dept. of Meteorology. URL http://www.o3d.org/abracco/Atlantic/Lorenz1956.pdf.

Lorenz, E., 1969: Atmospheric predictability as revealed by naturally occurring analogues. *Journal of the Atmospheric Sciences*, **26**, 636–646, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0469(1969)26{\%}3C636:APARBN{\%}3E2.0.CO;2.

Mariano, A., C. Costa, D. de Angelis, D. Pires Atala, F. Maugeri Filho, M. Wolf Maciel, and R. Maciel Filho, 2010: Genetic Algorithms (Binary and Real Codes) for the Optimisation of a Fermentation Process for Butanol Production. *International Journal of Chemical Reactor Engineering*, **8 (8)**, URL http://www.degruyter.com/view/j/ijcre.2010.8.1/ijcre.2010.8.1.2333/ijcre.2010.8.1.2333.xml.

Marty, R., I. Zin, C. Obled, G. Bontron, and A. Djerboua, 2012: Toward Real-Time Daily PQPF by an Analog Sorting Approach: Application to Flash-Flood Catchments. *Journal of Applied Meteorology and Climatology*, **51 (3)**, 505–520, doi:10.1175/JAMC-D-11-011.1, URL http://journals.ametsoc.org/doi/abs/10.1175/JAMC-D-11-011.1.

Michalewicz, Z., 1996: *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd ed., Springer-Verlag.

Murray, R., 1974: Indicators of monthly mean temperature and rainfall for England and Wales based on antecedent monthly pressure anomalies over the Northern hemisphere. *The Meteorological magazine*, **103**, 70.

Nap, J., H. van den Dool, and J. Oerlemans, 1981: A verification of monthly weather forecasts in the seventies. *Monthly Weather Review*, **109**, 306–312, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0493(1981)109{\%}3C0306{\%}3AAVOMWF{\%}3E2.0.CO{\%}3B2.

Nelder, J., and R. Mead, 1965: A simplex method for function minimization. *The Computer Journal*, **7 (4)**, 308–313, URL http://comjnl.oxfordjournals.org/content/7/4/308.short.

Obled, C., and W. Good, 1980: Recent developments of avalanche forecasting by discriminant analysis techniques: a methodological review and some applications to the Parsenn area (Davos, Switzerland). *Journal of Glaciology*, **25**, 315–346.

Radanovics, S., J.-P. Vidal, E. Sauquet, A. Ben Daoud, and G. Bontron, 2013: Optimising predictor domains for spatially coherent precipitation downscaling. *Hydrology and Earth System Sciences*, **17 (10)**, 4189–4208, doi:10.5194/hess-17-4189-2013, URL http://www.hydrol-earth-syst-sci.net/17/4189/2013/.

Radcliffe, N., 1991: Forma Analysis and Random Respectful Recombination. *Proceedings of the Fourth International Conference on Genetic Algorithms*, 222–229, URL http://stochasticsolutions.com/pdf/icga91.pdf.

Radinovic, D., 1975: An analogue method for weather forecasting using the 500/1000 mb relative topography. *Monthly Weather Review*, **103 (7)**, 639–649, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0493(1975)103{\%}3C0639{\%}3AAAMFWF{\%}3E2.0.CO{\%}3B2.

Rechenberg, I., 1973: *Evolutionsstrategie: Optimierung technisher Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog.

Schraudolph, N., and R. Belew, 1992: Dynamic parameter encoding for genetic algorithms. *Machine Learning*, **9 (1)**, 9–21, URL http://link.springer.com/article/10.1023/A:1022624728869.

Schwefel, H. P., 1981: *Numerical optimization of computer models*. John Wiley & Sons, Inc.

Schwefel, H.-P. P., 1993: *Evolution and Optimum Seeking*. John Wiley & Sons, Inc., New York, NY, USA.

Shabbar, A., and J. Knox, 1986: Monthly prediction by the analogue method. *Proceedings of the first WMO workshop on the diagnosis and prediction of monthly and seasonal atmospheric variations over the globe, Longe range forecasting Res. Rep. Ser. 6, Vol II, Tech. Doc. WMO/TD87*, 672–681.

Sievers, O., K. Fraedrich, and C. C. Raible, 2000: Self-Adapting Analog Ensemble Predictions of Tropical Cyclone Tracks. *Weather and Forecasting*, **15 (5)**, 623–629, doi: 10.1175/1520-0434(2000)015⟨0623:SAAEPO⟩2.0.CO;2, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0434{\%}282000{\%}29015{\%}3C0623{\%}3ASAAEPO{\%}3E2.0.CO{\%}3B2.

Smith, J., and T. Fogarty, 1997: Operator and parameter adaptation in genetic algorithms. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, **1 (2)**, 81–87, URL http://link.springer.com/article/10.1007/s005000050009.

Syswerda, G., 1989: Uniform Crossover in Genetic Algorithms. *Proceedings of the 3rd International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers Inc., 2–9.

Toth, Z., 1989: Long-range weather forecasting using an analog approach. *Journal of Climate*, **2**, 594–607, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0442(1989)002{\%}3C0594:LRWFUA{\%}3E2.0.CO{\%}3B2.

Woodcock, F., 1980: On the use of analogues to improve regression forecasts. *Monthly Weather Review*, **108 (3)**, 292–297, URL http://journals.ametsoc.org/doi/abs/10.1175/1520-0493(1980)108{\%}3C0292{\%}3AOTUOAT{\%}3E2.0.CO{\%}3B2.

Wright, A. H., 1991: Genetic Algorithms for Real Parameter Optimization. *Foundations of genetic algorithms*, **1**, 205–218, doi: 10.1016/B978-0-08-050684-5.50016-1.

Zitzler, E., M. Laumanns, and S. Bleuler, 2004: A tutorial on evolutionary multiobjective optimization. *Metaheuristics for Multiobjective Optimisation*, 3–37, URL http://link.springer.com/chapter/10.1007/978-3-642-17144-4{_}1.