Original paper Soft Computing 4 (2000) 68–75 © Springer-Verlag 2000

Multiple change-point detection with a genetic algorithm

A. Jann

Abstract A common change-point problem is considered where the population mean of a random variable is suspected of undergoing abrupt changes in course of a time series. It is usual in practice that no information on positions or number of such shifts is available beforehand. Finding the change points, i.e. the positions of the shifts, in such a situation is a delicate statistical problem since any considered sample may actually represent a mixture of two or more populations where values from both sides of a yet unrecognized change point are unconsciously assembled. If this is the case, underlying assumptions of an employed statistical two-sample test are usually violated. Consequently, no definite decision should be based on just one value of the test statistic. Such a value is rather, as a precaution, to be regarded as an only approximate indicator of the quality of a hypothesis about change-point positions. Given these conclusions, it is found imperative to treat the problem of multiple change-point detection as one of global optimization. A cost function is constructed in such a manner that the change-point configuration yielding the global optimum is compliant with statistical-theoretical requirements to the utmost extent. The used advanced optimization tool, a genetic algorithm, is both efficient – as it takes advantage of the information about promising change-point positions encountered in previously investigated trial configurations – and flexible (as it is open to any modification of the change-point configuration at any time). Experiments using numerical simulation confirm adequate performance of the method in an application where a common change-point detection procedure based on Student's two-sample *t*-test is used to detect an arbitrary number of shifts in the mean of a normally distributed random variable.

Key words Time series, Multiple change points, Global optimization, Genetic algorithm, *t*-test

1 Introduction

Locations where the probability distribution function abruptly changes in course of a time series are generally referred to as change points. The case which has attracted most attention is that of a population mean undergoing shifts at some points while remaining constant in-between those shifts. Figure 1 shows a simple example, where elements 1 to 10 originate from a normally distributed population with mean 0 and standard deviation 1 (shortly written as N(0,1)) whereas elements 11 to 20 are sampled from N(1.5,1). Since the term 'change point' designates the index of the last element before the mean undergoes a jump, the change point is located at position 10.

A respectable number of different two-sample tests are used for detecting the position of a single discontinuity in the mean; see for example the list compiled in [12] where also some hints concerning the performance of individual tests are given. For normally distributed variables, the location where the absolute value of Student's t is largest can be considered as the most likely position of a change point [3, 23]. The well-known t-test compares two samples of a normally distributed variable in order to find out whether these samples originate from the same population or not. The formula for t is

$$t = \frac{\mu_1 - \mu_2}{\sigma^* \sqrt{1/n_1 + 1/n_2}} \tag{1a}$$

with

$$\sigma^* = \sqrt{\frac{n_1\sigma_1^2 + n_2\sigma_2^2}{n_1 + n_2 - 2}} \ , \tag{1b}$$

where μ_1 and μ_2 are the two sample means, σ_1 and σ_2 are the respective standard deviations, n_1 and n_2 are the sample sizes. When the t-test is used as a change-point detector, the considered series is split at the individual observations, the basic statistical parameters μ and σ are computed for the resulting sub-samples, and t is derived. Furthermore, one can compute the associated cumulative Student's probability density function $\Phi(-|t|)$ with $n_1 + n_2 - 2$ degrees of freedom. The probability of equality of the compared sample means is easily derived via

$$s = P(\mu_1 = \mu_2) = 2\Phi(-|t|) \ . \tag{2}$$

A graph of s for the introductory example is included in Fig. 1. Since s is lowest (i.e. t is largest) when the series is split into two sub-samples of size 10, the most probable position of the change point is at no. 10 (which is in agreement with the design specifications of the series). Strictly speaking, however, this evaluation is not entirely compliant with the nature of the t-test since generally a sample from a single population is compared with a mixture of the two involved populations; the only exception is at position 10 where the tested position coincides with the actual change point. Fortunately, the mixing of

A. Jann
Central Institute for Meteorology and Geodynamics
Hohe Warte 38, A-1190 Vienna, Austria

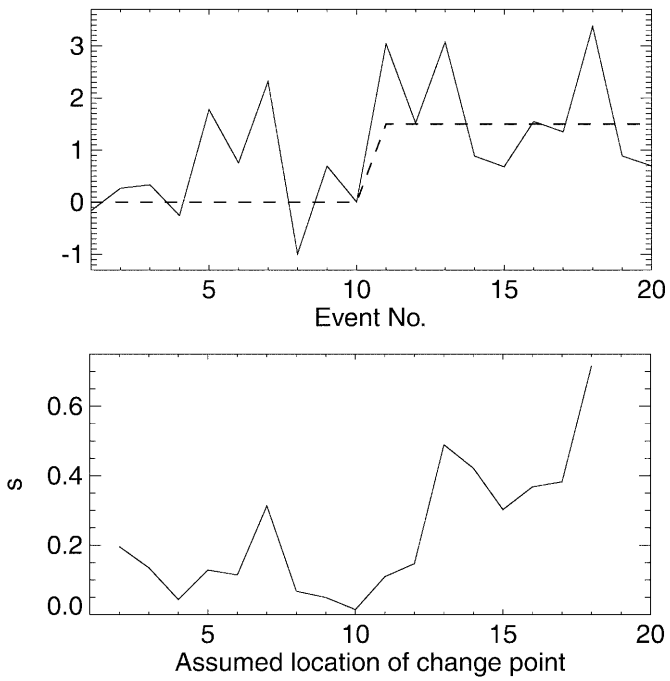

Fig. 1. Upper panel: Time series representing a prototype example of the considered change-point problem. The population mean (dashed line) undergoes a single shift from 0 to 1.5 after event no. 10. Lower panel: Corresponding graph of s (Eq. 2) as function of assumed change-point positions

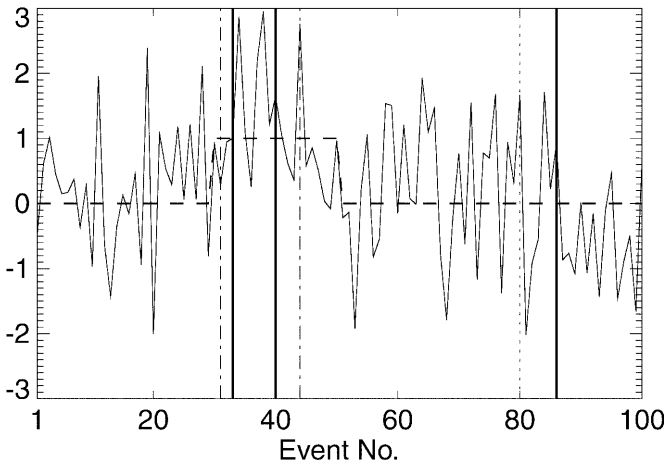

Fig. 2. Time series constructed from populations with N(1,1) (events 30 to 50) and N(0,1) (elsewhere), respectively; the dashed line delineates the course of the population means. Through a full combinatorial search, the optimum positions of 1, 2 or 3 change points were sought (criterion: minimization of $\bar{s}$ under the constraint that each s is lower than 0.01, cf. Eq. 2). The vertical dotted line marks the location of the single change point with the lowest s, positions of the best change-point pair are indicated by vertical dash-dotted lines, and the positions of the optimum change-point triplet are marked by the bold solid lines

populations tends to obscure the magnitude of a present single shift whereby the proper placement of the change point is supported.

Theoretical complexity increases and the consequences of deviations from statistical prerequisites become less clear if one has to expect numerous change points hidden somewhere in the series. The intuitive adaptation of the two-sample maximum-$|t|$ test to this situation is to find the change-point configuration optimizing some average function of all $|t|$-values computed for any two adjacent resultant sub-series. Since the distribution of t changes with the size of the sub-samples, it is impermissible to draw conclusions directly from t-values for different partitionings but one can compare $\Phi(-|t|)$ or s, respectively. The probabilities $s_j = P(\mu_j = \mu_{j+1})$ are therefore averaged over j (running from 1 to the number of assumed change points) to give $\bar{s} = 2\Phi(-|t|)$. Figure 2 provides the analysis of a series with two artificially imposed change points. If a restriction to a single change point is made for the maximum-t test, the most probable position of the discontinuity is at event 80. Searching for two significant shifts (significance criterion: s for two separated sub-series less than 0.01), the criterion of minimum $\bar{s}$ yields change points at events 31 and 44. The maximum number of significant shifts that could be found were three, with change points being optimally placed at events 33, 40 and 86. Two aspects are noteworthy:

(1) As long as a single significant discontinuity is undetected, values of the test statistic are distorted so that the precise location of any other discontinuity cannot be given.

(2) But even then, the indicated change points are quite close to change-point positions appearing in the optimum configuration with the largest number of shifts.

These observations can be reformulated to express demands on algorithms used for change-point detection:

(1*) The algorithm should as long as possible be susceptible to shifts in change-point positions.

(2*) The method should nevertheless efficiently make use of the information on interesting change-point positions that has been gained during preceding evaluations of candidate configurations.

The normal situation in a practical application is that nothing can be safely assumed at the beginning of the analysis concerning the number or positions of present change points. Therefore each inspected sample potentially is a mixture of populations. Imagining the various possible compositions of samples, it becomes evident that it certainly is an arduous task to lay a universal theoretical foundation for the identification of multiple change points. One may perhaps argue that the challenging open theoretical questions (see also [7] in this context) should be satisfactorily resolved before any practical solution is envisaged. On the other hand, change-point analyses are of interest in disciplines like reliability analysis, analysis of mortality data, baseball (see articles in [22]), hydrology [3, 4] and climatology (see the recent review of [17]) where a somewhat incomplete theoretical-mathematical foundation of a solution method should often be acceptable as long as empirical evidence of the proper performance can be provided. It is intended here to assume such a pragmatic standpoint. From what was said above, it appears imprudent to entirely disregard the information content of computed quantities even when one is almost sure that the

test has erroneously been applied to a mixture of populations. Retaining such information means accepting temporary violation of statistical theory. The crucial question is how to handle intermediate results so that they guide to the true solution despite their imperfect character. The attractive feature of the approach introduced hereafter is that it enables the construction of a numerical change-point detector based on the cautious point of view that numerical values of a quantity like s are in general only approximate indicators of the quality of hypotheses about change-point positions.

2
Previous approaches to multiple change-point detection

Several techniques to exploit two-sample tests for multiple change-point detection have already been developed. Some authors [1, 18] remarked that graphs of the statistics they used were capable of indicating the position of a larger number of change points even though the statistical parameters had been derived for the case of a single change point. This eventually led to a subjective search for the positions of the change points, with an obvious technical resemblance to the evaluation of cumulative sum plots in [3], [14] or [20].

A couple of authors attempted to use two-sample tests in some iterative way to detect the presence and locations of multiple change points objectively. In [11] and [15], the series was adjusted for the associated shift in μ after a change point was detected and the scheme proceeded in investigating the adjusted series. Other authors split the series after detection of a change point and repeated the test on the resulting sub-series [8, 16]. Schemes of this type, where the number of change points is increased steadily, will be referred to as sequential schemes in the following (they are not to be confused with sequential schemes of another type – which is not covered here since we deal only with the retrospective analysis of time series – where a test statistic is continuously updated with the information from additional observations becoming available with time; for an application-motivated example see the adaptation of the Kalman filter in [24]).

The problematic aspect of sequential schemes was already touched upon in Fig. 2. Since a practical procedure for multiple change-point detection cannot avoid the situation where certain statistical quantities are computed though not all prerequisites are met, procedures which rely on irrevocable hypotheses prescribed in the very early stages are risky. For the case of Fig. 2, after identifying the optimum single change point at event no. 80 (on the basis of a rather dubious value of s) and restricting further search to configurations with a change point at that position, it has already become impossible to reach the global optimum with three change points. The best position of a second change point is then found to be at event no. 20. Fixing the two positions and searching for a third change point identifies no. 44 as the optimum position. In view of the globally optimum configuration with positions at events 33, 40 and 86, there clearly is room for improvement. Further experimental evidence of the limitations of sequential schemes is provided by the results of [6]. The test used there was employed in two different sequential modes: Consideration of all configurations with one change point added to the current working hypothesis (mode A) or consideration of all configurations with either one or two additional change points (mode B). It was concluded that for the general case, mode A is unsatisfactory as it failed to detect change points which were in fact found with mode B. This indicates that it is important in sequential procedures to guess the right number of additional change points which have to be introduced at one step. It suffices to look at the discussion on Fig. 2 to anticipate that there are configurations where it is beneficial to introduce even three or more new change points simultaneously. Eventually, a full combinatorial search is demanded. However, through increasing the length of the time series, it is easy to make a combinatorial investigation of all possible change-point configurations computationally prohibitive for any present and future computer generation. Fortunately, genetic algorithms represent a valuable remedy in that they can efficiently produce good optimization results for combinatorial problems without scanning the whole variety of conceivable options. The following sections investigate this approach.

3
The algorithm

3.1
Binary coding

In some previous applications of genetic algorithms, the problem was transformed to a binary representation first (e.g. [10, 21]). This is not obligatory and different coding has already been used successfully [9, 25]. However, since the present problem is inherently one of the binary type, namely a yes/no-decision concerning the presence of a change point for each element of the series, the traditional binary method will be employed. The transition of change-point hypotheses to auxiliary bit sequences of zeros and ones is straightforward by defining that the symbol '1' should signify a change point. For example, for the 20-element series of Fig. 1 with its change point at event 10, the solution is described by the sequence 00000000010000000000. In the next section, a cost function will be defined that evaluates the adequacy of hypotheses about change points with respect to the considered time series. The exercise is to find the bit sequence minimizing this cost function.

3.2
The cost function

For analyzing the series of Fig. 2, it was already operated with the expression $\overline{\Phi(-|t|)}$ in order to find an optimum configuration of a pre-specified number of change points. For the general problem with an unknown number of change points, the more universal cost function which was chosen to be minimized writes

$$C = \overline{\Phi(-|t|)} - \beta_1 vb - \beta_2 b \quad . \tag{3}$$

The term $-\beta_1 vb$ reflects the wish to identify all potential change points. The symbol v designates the number of indicated change points; β_1 is set to 1 if all assumed change

points satisfy a prescribed significance criterion $s \leq s_t$ (or, equivalently, $\Phi(-|t|) \leq s_t/2$), otherwise $\beta_1 = 0$. Choosing the factor b such that it satisfies $b \geq s_t/2$ ensures that a configuration with $v + 1$ significant change points (and no change point with significance below the specified level) is always favoured in comparison with a counterpart with only v change points. Thus, the identification of the maximum number of discontinuities is supported whereas it is considered a subordinate (but nevertheless important) task to search for the best solution among the candidate configurations with that maximum number of significant jumps in the mean. The second bonus term $-\beta_2 b$ allows to distinctly separate the bit sequences which contain some interesting change-point positions from those seemingly without any. This is accomplished simply through setting β_2 to 1 when at least one change point meets the significance criterion, $\beta_2 = 0$ otherwise. The bonus concept in the presented form leads to the somewhat unusual occurrence of negative cost functions but this is of course merely a cosmetic aspect. The same applies to the fact that for $b > s_t/2$, the range of C shows gaps as certain values of C cannot be assumed.

Taking into account the previously highlighted fact that the application of the chosen statistical test may suffer from a serious flaw as long as one or more of the actual change points have not yet been identified, the contributions of the last two terms on the right-hand side of (3) can also be viewed as statistical bonus, awarded for the higher certainty that the t-test is properly used when more substantial shifts are already identified.

A classification of bit sequences comprises the following groups:

1) The sequences containing entries of '1' but not a single significant change point: positive values of C;
2) those with both significant and insignificant change points: $C > -b$;
3) the null sequence without change point, for which we define $C = -b$, thus completing definition (3);
4) the sequences where each entry of '1' marks a significant change point: $C < -b$.

The bit sequence optimizing cost function (3) belongs to one of the last two groups, of course.

Strictly speaking, there was a fifth group of bit sequences: those which were excluded from the competition after lower limits were imposed on the distance between assumed change points. There is the finding in [13] that at least five data items of each involved population are required to attain sound estimates of the mean. Easterling and Peterson [8] also forced two adjacent change points to be at least five events apart for statistical reasons. Likelihood ratio tests moreover have the undesired tendency to favour the placement of change points at the ends of the series, thereby cutting off a small unrepresentative sample ([4]; the t-test can be considered a special form of likelihood ratio test, cf. [3, 23]). Hence there is enough motivation to ignore hypotheses resulting in one or more separated sub-series with less than five elements. No further initial assumption on number or positions of change points was made since – as already mentioned – knowledge about these factors is scarce in practice.

3.3
The genetic algorithm

In the initial step of the computations, a certain number Q of parental bit sequences were produced ($Q = 200$ throughout the paper). A tenth of the bits were randomly selected and set to 1. Bits were altered from 1 to 0 if one of the four preceding bits was 1, i.e. from two change-point indicators separated by less than four zeros, only the first was retained in order to avoid too short sub-series, as motivated above. After creating the initial hypotheses on change-point positions in that way and computing the cost functions C of the parents, the selection step was carried out where Q surviving candidates were selected from the parent generation, with probabilities being proportional to

$$P_j = \alpha C_{\max} - C_j \qquad (4)$$

(identical with the linear approach of [21]). The parameter α (≥ 1) determines the importance ascribed to differences in C. With increasing α, the numbers P_j become more and more similar to each other and the selection step thus becomes less discriminative. For reasonable values of α, parents with low C may be selected twice or even more often whereas candidates with high C have a small probability to ever be selected so that they are likely to be withdrawn from the competition. For $\alpha = 1$, the worst candidate of the parent generation ($C = C_{\max}$) even has a selection probability of 0.

In the subsequent crossover step, the selected parents were paired and subject to an exchange of sequences with the respective companion. For example, from the parents

A: 000001000000000000001000000 and
B: 000000000000000000100000000,

a crossing at, say, bit no. 12 yields

A_C: 000001000000000000100000000 and
B_C: 000000000000000000001000000,

whereas crossing at the edges would leave the parents unaltered. The positions where parents were split were individually determined for each pair from a uniformly distributed random variable. The crossing took place strictly in the order in which parents were selected in the preceding step, i.e. the first was paired with the second, etc., regardless of whether two identical strings were paired or of other 'pathological' conditions which could have been checked.

Those parents which had survived the selection step were also subject to a mutation step where the values of some randomly selected bits were altered. In our implementation, the number of mutations was chosen to be Q, i.e. on average each string underwent a single change. While crossover is an effective means of combining previously created promising partial sequences, it is not too productive of options with a larger number of change points. In order to introduce a complementary mechanism, $3Q/4$ mutations were reserved to take place from 0 to 1 whereas only $Q/4$ mutations were changes from 1 to 0 provided, of course, that a sufficient number of change points were present in the considered population. A result for the above parents A and B may look as follows:

A_M: 00000000000000001001000000 (two mutations, one from 0 to 1, the other from 1 to 0),
B_M: 00000000100000000100000000.

In the first iteration after the finding of a new best sequence, the mutation step was different in order to scrutinize sequences closely resembling the current optimum which one would be inclined to consider the best basis for further progress. The effect of introducing one additional change point was investigated for each possible position. The second mechanism producing alternative bit sequences was to shift each of the indicated change points. Assuming that parent A is a freshly created 'best sequence', we obtain:

* 10000100000000000001000000, 01000100000000000001000000, ... through adding a change point,
* 10000000000000000001000000, 01000000000000000001000000, ... by shifting the left change point and, finally,
* 10000100000000000000000000, 01000100000000000000000000, ... by shifting the right change point of sequence A.

Cost functions C were calculated for both the crossover population and the mutated sequences, and the best $Q/2$ unique representatives of each group were elected to enter the next parent generation (which was then subject to the selection step, and so forth). Undesired sequences with change points being separated by less than 4 zeros could be produced through crossover or mutation (like, for example, the above sequence A_M) but such sequences were immediately excluded from any further processing.

It is an interesting aspect for software development that particularly in the later stages of the procedure, previously investigated bit sequences frequently reappear again and again. It was found economic to store C for all investigated sequences and first scan that collection instead of blindly entering a repeated computation of C for one and the same bit sequence. The second purpose of this archive was that randomly chosen bit sequences from it were added to complete the new parent generation when it was not possible to find Q suitable representatives after mutation and crossover (either because there were too many illegal sequences or not enough mutually different candidates).

A noticeable positive impact was obtained through the introduction of a two-step procedure where initially the significance limit was relaxed. After convergence for a higher value of s_t, the Q best bit sequences (as judged for the eventually desired stricter limit) were taken as parent generation and the procedure was continued, now with the tighter threshold. All runs described in the following section started with $s_t = 0.05$ and then switched to a significance threshold of $s_t = 0.01$. While any change point of the solution for $s_t = 0.01$ does of course satisfy $s \leq 0.05$, the converse need not hold. A change point with significance between 95 and 99 per cent may influence the course of the search such that the algorithm misses the solution for $s_t = 0.01$; therefore the second run with the tighter limit is necessary unless all change points of the solution for $s_t = 0.05$ satisfy the $s \leq 0.01$-criterion as well. The temporarily lower requirements help to tackle difficult situations where, for example, change points need to be placed at their true locations in order to satisfy $s \leq 0.01$ while through a deviation by one position, s fails to reach that limit. Bit sequences resembling the true solution are more likely to be retained for the relaxed significance limit due to additional contributions of the bonus terms to C. Through the mechanisms that search among similar bit sequences, the solution which only just satisfies the $s \leq 0.01$-criterion can then be found more easily. The additional computational effort is small since $\Phi(-|t|)$ does not depend on the significance level s_t; only the bonus terms have to be evaluated twice. This resulted, of course, in a practical implementation where $C(s_t = 0.05)$ and $C(s_t = 0.01)$ were simultaneously computed (and stored in the computer's virtual memory) when a bit sequence was encountered during the run with the threshold $s_t = 0.05$.

As a rule, it was found that not much experimentation is necessary to find proper values for α, b, Q and the mutation rate. Inappropriate assumptions were readily identified through a wide scatter in the ostensibly optimum bit sequences of repeated runs. Moreover, the range of reasonable values was generally a fairly broad one, with the genetic algorithm exhibiting robustness even against major changes such as an increase from $\alpha = 1.0$ to $\alpha = 2.0$. Sambridge and Drijkoningen [21] similarly noted the user-friendly behaviour of genetic algorithms which allows the achievement of good results with only little preparatory work. Results will be reported for runs with $\alpha = 1.5$; for b, a value of 0.05 was found adequate for both used significance thresholds $s_t = 0.05$ and $s_t = 0.01$. The ordering of bit sequences with respect to their cost functions does not depend on the actually chosen value of b, provided $b \geq s_t/2$, but the convergence speed of the computer program can be severely affected (determining the risk that the program is terminated too early because of presumed convergence). With b too small, poor bit sequences are not disfavoured strongly enough; with b too large, one retains only the very best sequences but neglects interesting material contained in medium sequences. There is thus an apparent similarity to the function of α.

Monitoring of the convergence led to the conclusion that the procedure with the chosen set of parameters could be confidently terminated when there were 30 consecutive iterations (consisting of selection, crossover, and mutation) without any progress in lowering the minimum C. This therefore became the stopping rule for the experiments documented hereafter.

4
Numerical experiments

The developed algorithm was evaluated using numerical simulation. The basic sample consisted of 100 series, each comprising 100 computer-generated random numbers. The specifications were that the values should form a normal distribution with mean 0 and standard deviation 1. The initial experiment (section 4.1) examined these 100 homogeneous series with constant population mean $\mu = 0$. Each series was then modified through adding an increment $\Delta\mu$ to its elements no. 30 to 50, i.e. two change points were imposed at events no. 29 and 50 where the

distribution first changed to N(Δμ, 1) and returned to N(0,1) later (Note: Fig. 2 is based on such a series). Results for the inhomogeneous series are provided in section 4.2.

4.1 Homogeneous series

For the homogeneous series there should be just one series in 100 where, for example, the first 5 and the remaining 95 elements show significantly different means, according to the prescribed threshold 0.01 for the significance level s_t. There is, however, likewise a 1 per cent chance that a 95-5 separation accidentally looks inhomogeneous. The probabilities are fortunately not linearly accumulative since, e.g., homogeneity of a 50-50 separation is unlikely to turn into inhomogeneity in a 48-52 separation. Clearly, however, one must expect that considerably more than 1 per cent of the series are found inhomogeneous. (There is moreover the possibility that a sub-series in the middle appears to have a different population mean than the two adjacent counterparts, so the statistical problem eventually becomes quite intricate.) A high false alarm rate was in fact diagnosed when the genetic algorithm was applied to the homogeneous series. Only 58 series exhibited the outcome of no discontinuity at the $s \leq 0.01$-level while the number of series with 1, 2, 3, 4 or 5 indicated change points was 5, 19, 5, 10 and 3. Short sub-series containing rather small or large values of the population are for trivial reasons located more often in the middle than at the edge of the series which can well explain why pairs of change points had to be found more frequently than solitary change points.

For series with more than one significant change point, undetected shifts may obscure the significance of the other shifts. The challenge can be a huge one: Our most extreme case was a series where for the assumption of a single change point, s was never smaller than 0.254 and thus far less significant than required. The genetic algorithm revealed two change points where, according to the t-test, the statistical probability that the separated samples belong to the same population was 0.75% for the first change point and 0.31% for the other. Among the 100 investigated homogeneous series, only 11 had a minimum s below 0.01 for the assumption of a single change point, i.e. 89 series would have appeared as homogeneous at first glance. This is to be compared with the outcome that the genetic algorithm eventually found statistically significant jumps in the mean for 42 series. If we had used a sequential procedure, values of s would have been the only reasonable basis to formulate a stopping criterion. The misleading values of s then could have easily led to frequent premature termination of the search. Though the result of homogeneity were correct in the sense of the desired outcome, it is not satisfactory when, on the other hand, significant discontinuities are revealed by the genetic algorithm.

In order to gain further information on the quality of the genetic algorithm, a combinatorial-type investigation was launched. With the available equipment, it was found a bearable effort to check every configuration with three or less change points. Since the genetic algorithm indicated the presence of more than three change points only for 13 series, a quite complete comparison between the combinatorial optimum and the result of the genetic algorithm was possible. There was a single case where the change points were slightly displaced (by 2 and 7 positions, respectively). In no other case, the genetic algorithm could be proven wrong through the combinatorial search.

The number of investigated different bit sequences that were examined during the investigation of a series ranged from 3920 to 11351, with about 6700 on the average and only four examples where this number exceeded 10000. There is no reason to expect significantly different figures for cases with imposed artificial discontinuitites. In fact, for the experiments discussed in the next sub-section similar values were observed; since they were even somewhat lower, the computation of C for about 5700 sequences can be recorded as average effort. The genetic algorithm achieved the qualitative result of inhomogeneity quite quickly as, if existing at all, series with a cost function below $-b$ were generally created already during the third or fourth iteration. Usually, the final number of change points was indicated soon afterwards. Ultimate termination admittedly occurred considerably later. However, compared with the complete investigation of configurations with up to three change points, the genetic algorithm was able to compete with the combinatorial method in terms of computing time. For combinatorial computations extended to allow for a fourth change point, a ratio of 1/50 in favour of the genetic algorithm can be roughly estimated.

4.2 Series with imposed change points

Let us now consider those examples where two change points were artificially introduced by adding Δμ to elements no. 30 to 50 of the previously considered homogeneous series. Results are presented for experiments with Δμ = 1.0 and 1.5, respectively. Recall that the standard deviation of the populations, σ, was 1. For the run with Δμ = 1.0 (1.5), there were 73 (65) cases with three or less change points indicated by the genetic algorithm. For them, a comparison with the same (incomplete) combinatorial search as used above was carried out. The diagnosis reads as follows:

- Δμ = 1.0: In one case, only two change points were found by the genetic algorithm though a solution with three significant change points existed. For the other 72 cases, the solutions were equal.
- Δμ = 1.5: There were three cases where solutions with three significant change points existed but only two change points were found by the genetic algorithm. For three series, the number of change points was correct (namely 3) but slight displacements were observed in the location of two or three of the change points. For the remaining 59 cases, the solutions were equal.

One property that could indicate failure of a concept is lack of reproducibility. Repeating the experiments yielded identical results in 87 (Δμ = 1.0) and 85 (Δμ = 1.5) cases. Differences between parallel runs generally affected the

cases with numerous (≥3) discontinuities where in any case at least two change points were indicated. It appears that the genetic algorithm actually never failed to qualitatively detect the two imposed signals of magnitude σ and 1.5σ, respectively, but suffered from occasional problems to detect random-noise signals of smaller magnitude and less importance. While it is the major advantage of a genetic algorithm that not the entire pattern space needs to be scanned to find a (near-)optimum solution, this is of course also associated with some risk. The algorithm is after all a random process and can therefore occasionally overlook the global optimum if the generated random numbers never lead to the best configuration. It would hence be unrealistic to expect that – under the side–condition that the computational efficiency be retained – the quite satisfactory figures around 85 per cent can somehow be improved to a reproducibility near 100 per cent for a genetic algorithm. Nevertheless, in order to increase the probability that the global optimum is hit, it is a good idea to run the procedure twice to give the algorithm the chance to approach the true solution on a fresh random path.

The frequency distributions of the indicated positions of change points are shown in Fig. 3. For properly working methods, peaks around the positions of imposed change points should be observed, becoming sharper for larger $\Delta\mu$. Because of random noise, a certain scatter around the intended positions but also some additional signals have to be accepted. The patterns observed for the genetic algorithm approach are therefore absolutely normal.

5 Discussion

A method of extending a two-sample test to a procedure dealing with multiple change points has been sought. It was found imperative to consider the problem as one of global optimization after it had been recognized that use of two-sample tests is inevitably associated with the occurrence of dubious values of the test statistic. An objective cost function was defined such that its global minimum should be assumed by a partitioning of the series being statistically correct to the utmost extent. The present study suggests a genetic algorithm as adequate and efficient tool accomplishing the desired global optimization.

Requirements to avoid the identified potential negative implications of irrevocable conclusions based on a questionable value of s were formulated in the introduction. The proposed genetic algorithm satisfies them by not restricting itself to material from a single bit sequence but holding in memory all the change-point positions where cost functions noticeably fall below average. Secondly, alternatives in change-point positions are constantly produced owing to the mutation component. The right balance between retention of interesting configurations, on one hand, and innovation, on the other, can be achieved through tuning of parameters like mutation rate, α, etc.

Confidence in the genetic algorithm's proper performance was gained here through comparison with combinatorial-type runs and through observed repeatability. The validity of the concept was shown with help of numerically simulated data where it was guaranteed that the data suited the nature of the chosen test, the t-test. The t-test merely is the most familiar procedure used for change-point detection which was the motivation why it was selected here. It is not an integral component of the genetic algorithm strategy, though. An obvious limitation of the t-test is its parametric nature so that the procedure is not appropriate for each physical variable and additionally not as robust as sometimes necessary in the presence of outliers (see [15]). Moreover, similar but more powerful two-sample tests have already been developed ([1, 18]; statement of quality according to [8]). It is also undoubted that autocorrelation or trend in the data often need to be taken into account either by using appropriate tests, e.g. those of [2] or [4], or through preceding treatment of the series as in [5] or [26]. The potential modifications result only in a substitution of $\Phi(-|t|)$ in cost function (3) by another, equivalent, statistical quantity while the optimization part remains unaffected. Therefore, after the basic statistical test fitting the character of the data is introduced, the genetic algorithm should show the same satisfactory performance as it was observed here.

Besides a change in the statistical test, application to real-world data is likely to be accompanied also by a change of view concerning the change-point signals being due to omnipresent random noise. Many statements were based here on the standpoint that identification of a

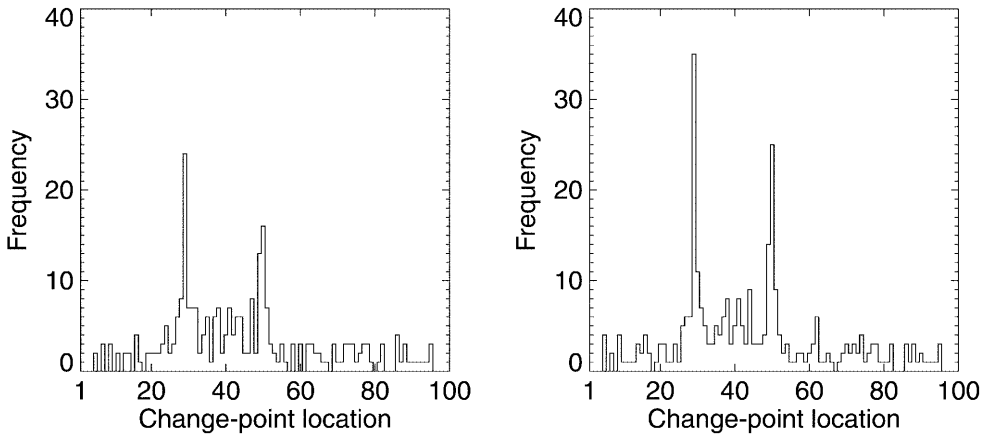

Fig. 3. Frequency distributions of change-point locations as indicated by the genetic algorithm, for the two experiments with artificially imposed change points. According to the experiment specifications, the true locations of the change points are at no. 29 and 50 (where actually the highest peaks of the histograms are located). Left: For the experiment with $\Delta\mu = 1.0$. Right: For the experiment with $\Delta\mu = 1.5$

partition with only significant t-values is nothing else than an admissible result of the optimization of (3). The more change points are identified the better. By virtue of the established statistical significance, there was no basis to judge any detected change point as an error of the genetic algorithm. However, it is a matter of concern in an applied science when a large number of signals occur for no other reason than random noise. Experiments with $s_t = 0.01$ showing 113 'unintended' change-point signals in 100 series (cf. section 4.1) call for remedies, e.g., in climatology where significance levels of 1% [15] or even 5% [19] and 10% [11] are indeed used. The revision of a statistically correct decision is necessarily empirical (examples: [12, 17]), depends on the application, and is separated from the change-point detection as such. Though this post-processing hence need not be subject of this paper, it must at least be pointed out that the interaction with the genetic algorithm should take place in an iterative framework. After rejection of an indicated change point, the genetic algorithm is to be consulted again in order to find the optimum bit sequence not having a '1' at the rejected position. This second run should vastly benefit from the archived information collected during the initial, unconstrained run. For the then obtained solution, one again decides whether the change-point signals can be accepted or not, and so forth.

References

1. **Alexandersson H** (1986) A homogeneity test applied to precipitation data, J Climatol, 6, 661–675
2. **Alexandersson H, Moberg A** (1997) Homogenization of Swedish temperature data. Part I. Homogeneity test for linear trends, Int J Climatol, 17, 25–34
3. **Buishand TA** (1982) Some methods for testing the homogeneity of rainfall records, J Hydrol, 58, 11–27
4. **Buishand TA** (1984) Tests for detecting a shift in the mean of hydrological time series, J Hydrol, 73, 51–69
5. **Busuioc A, von Storch H** (1996) Changes in winter precipitation in Romania and its relation to the large-scale circulation, Tellus, 48A, 538–552
6. **Caussinus H, Mestre O** (1997) New mathematical tools and methodologies for relative homogeneity testing, In: Proceedings of the first seminar for homogenization of surface climatological data, Budapest, Hungary, 6–12 October 1996, pp 63–82
7. **Dragalin V** (1996) The retrospective change point problem, Econ Qual Control, 11, 3–22
8. **Easterling DR, Peterson TC** (1995) A new method for detecting undocumented discontinuities in climatological time series, Int J Climatol, 15, 369–377
9. **Franchini M** (1996) Use of a genetic algorithm combined with a local search method for the automatic calibration of conceptual rainfall-runoff models, Hydrol Sci J, 41, 21–39
10. **Gallagher K, Sambridge M, Drijkoningen G** (1991) Genetic algorithms: an evolution from Monte Carlo methods for strongly non-linear geophysical optimization problems, Geophys Res Lett, 18, 2177–2180
11. **Herzog J, Müller-Westermeier G** (1997) Homogenization of various climatological parameters in the German weather service, In: Proceedings of the first seminar for homogenization of surface climatological data, Budapest, Hungary, 6–12 October 1996, pp 101–111
12. **Herzog J, Müller-Westermeier G** (1998) Homogenitätsprüfung und Homogenisierung klimatologischer Meßreihen im Deutschen Wetterdienst (in German), Berichte des Deutschen Wetterdienstes, no. 202. Offenbach am Main: Selbstverlag des Deutschen Wetterdienstes
13. **Karl TR, Williams CN Jr** (1987) An approach to adjusting climatological time series for discontinuous inhomogeneities, J Clim Appl Meteor, 26, 1744–1763
14. **Kohler MA** (1949) On the use of double-mass analysis for testing the consistency of meteorological records and for making required adjustments, Bull Amer Meteor Soc, 30, 188–189
15. **Lanzante JR** (1996) Resistant, robust and non-parametric techniques for the analysis of climate data: theory and examples, including applications to historical radiosonde station data, Int J Climatol, 16, 1197–1226
16. **Moberg A, Alexandersson H** (1997) Homogenization of Swedish temperature data. Part II. Homogenized gridded air temperature compared with a subset of global gridded air temperature since 1861, Int J Climatol, 17, 35–54
17. **Peterson TC, Easterling DR, Karl TR, Groisman P, Nicholls N, Plummer N, Torok S, Auer I, Böhm R, Gullett D, Vincent L, Heino R, Tuomenvirta H, Mestre O, Szentimrey T, Salinger J, Førland EJ, Hanssen-Bauer I, Alexandersson H, Jones P, Parker D** (1998) Homogeneity adjustments of in situ atmospheric climate data: a review, Int J Climatol, 18, 1493–1517
18. **Potter KW** (1981) Illustration of a new test for detecting a shift in mean in precipitation series, Mon Wea Rev, 109, 2040–2045
19. **Rhoades DA, Neill WA** (1995) The effect of station drift on the estimation of shifts in meteorological records due to site changes, Int J Climatol, 15, 207–220
20. **Rhoades DA, Salinger MJ** (1993) Adjustment of temperature and rainfall records for site changes, Int J Climatol, 13, 899–913
21. **Sambridge M, Drijkoningen G** (1992) Genetic algorithms in seismic waveform inversion, Geophys J Int, 109, 323–342
22. **Sinha B, Rukhin A, Ahsannullah M** (Eds) (1995) Applied change point problems in statistics. Proceedings of an applied change point conference, Baltimore, MD, 17–18 March , 1993
23. **Szinell C** (1997) Methods for homogenization of data series, In: Proceedings of the first seminar for homogenization of surface climatological data, Budapest, Hungary, 6–12 October 1996, pp 9–17
24. **Whittaker J, Frühwirth-Schnatter S** (1994) A dynamic changepoint model for detecting the onset of growth in bacteriological infections, Appl Statist, 43, 625–640
25. **Wilson WG, Vasudevan K** (1991) Application of the genetic algorithm to residual statics estimation, Geophys Res Lett, 18, 2181–2184
26. **Zwiers FW, von Storch H** (1995) Taking serial correlation into account in tests of the mean, J Climate, 8, 336–351