Earth and Space Science

RESEARCH ARTICLE

10.1029/2020EA001515

Key Points:

- A novel multiple-point statistics algorithm allowing to account for inequality constraints is proposed
- The method extends the capability of the direct sampling algorithm
- The method is illustrated for the simulation of elevation models in the central part of Switzerland

Correspondence to:

J. Straubhaar,
julien.straubhaar@unine.ch

Citation:

Straubhaar, J., & Renard, P. (2021). Conditioning multiple-point statistics simulation to inequality data. *Earth and Space Science*, *8*, e2020EA001515. https://doi.org/10.1029/2020EA001515

Received 16 OCT 2020
Accepted 13 FEB 2021

Conditioning Multiple-Point Statistics Simulation to Inequality Data

Julien Straubhaar[1] **and Philippe Renard**[1,2]

[1]Centre for Hydrogeology and Geothermics (CHYN), University of Neuchâtel, Neuchâtel, Switzerland, [2]Department of Geosciences, University of Oslo, Oslo, Norway

Abstract Stochastic modeling is often employed in environmental sciences for the analysis and understanding of complex systems. For example, random fields are key components in uncertainty analysis or Bayesian inverse modeling. Multiple-point statistics (MPS) provides efficient simulation tools for simulating fields reproducing the spatial statistics depicted in a training image (TI), while accounting for local or block conditioning data. Among MPS methods, the direct sampling algorithm is a flexible pixel-based technique that consists in first assigning the conditioning data values (so-called hard data) in the simulation grid, and then in populating the rest of the simulation domain in a random order by successively pasting a value from a TI cell sharing a similar pattern. In this study, an extension of the direct sampling method is proposed to account for inequality data, that is, constraints in given cells consisting of lower and/or upper bounds for the simulated values. Indeed, inequality data are often available in practice. The new approach involves the adaptation of the distance used to compare and evaluate the match between two patterns to account for such constraints. The proposed method, implemented in the DeeSse code, allows generating random fields both reflecting the spatial statistics of the TI and honoring the inequality constraints. Finally examples of topography simulations illustrate and show the capabilities of the proposed method.

1. Introduction

The need of stochastic simulation of spatial variables often arises in earth and environmental sciences, for understanding, making forecasts with uncertainty quantification and taking decisions related to natural and physical processes (Chiles & Delfiner, 2012). For example, Bayesian inverse modeling is a standard problem in groundwater hydrology in which there is a need to express prior information (Linde et al., 2015). In this context, one can use multiple-point statistics (MPS) to express the prior knowledge using a training image (TI) that reflects the expected spatial patterns resulting from geological processes.

Numerous MPS techniques have been developed since the early 2000s (Mariethoz & Caers, 2014), each one having their own level of complexity, advantages, and drawbacks. Conditioning realizations to hard data, that is, known values at given points, is an essential features of such methods. Whereas conditioning to hard data may be difficult for some patch-based algorithms, honoring hard data is straightforward for pixel-based methods, for example, *snesim* (Strebelle, 2002) or *Impala* (Straubhaar et al., 2013), because one cell is simulated at a time. The direct sampling strategy (Mariethoz et al., 2010), belonging to the second category, is one of the most flexible MPS algorithm. It allows for uni- and multi-variate simulations of categorical and continuous variables, and notably handles smooth changes of orientations and scaling, nonstationary TIs, proportion (Mariethoz et al., 2015), and connectivity constraints.

To our knowledge, no MPS method directly deals with inequality data, consisting in imposing lower and/or upper bounds at given locations. However, this type of information is frequently available in real applications. Indeed, the value of a physical parameter in some location could be approximately known, due to the low precision of a measuring instrument or to the conditions under which the measures have been taken. More generally, it makes sense to impose intervals of values around the measurements instead of the measured values themselves, because any measurement is subject to uncertainty. In some situations, only minimal or maximal bounds constrain a model. Considering for instance the simulation of the elevation of a stratigraphic unit over a bi-dimensional domain, a borehole not reaching the considered unit leads to a maximal value for the elevation at the borehole location. On the opposite, the observation of a formation older than the one of interest in a borehole or outcrop may indicate that the considered unit was eroded, and provides a minimal bound for the elevation of the base of this unit.

Inequality constraints can be directly imposed in the simulation or estimation of Gaussian processes by considering truncated multi-variate Gaussian distributions (Abrahamsen & Benth, 2001; Freulon & de Fouquet, 1993; Maatouk & Bay, 2017; Michalak, 2008). One technique to impose the inequality constraints is to use a Gibbs sampler (see e.g., Robert, 1995). Nevertheless, a Gaussian process is based on a covariance or variogram model that may be insufficient to model complex heterogeneous spatial structures.

In this study, we propose a solution to handle inequality data in the direct sampling MPS algorithm. This is done by accounting for possible lower and/or upper bound in the pattern around the simulated cell and searching for a compatible pattern within the TI. The pattern retrieved from the simulation grid (SG) is compared to the patterns scanned in the TI by computing a distance adapted to the situation where a pattern contains one or several nodes informed only by inequality constraints. This method is developed and implemented in the DeeSse code (Straubhaar, 2020), which then allows for a proper reproduction of the spatial statistics present in the TI while honoring hard and inequality data.

The study is organized as follows. The theory of the proposed method is explained in Section 2. Applications and examples are presented in Section 3, and finally, conclusions are given in Section 4.

2. Methodology—Direct Sampling With Inequality Data

The goal is to generate realizations of a continuous variable in a SG that reproduce the spatial statistics present in the TI and that honor a conditioning data set.

2.1. Type of Conditioning Data

A conditioning data set consists in an ensemble of **hard data points** (x, z), where x is a cell in the SG and z is a value of the simulated variable Z at x, and an ensemble of **inequality data points** $(x, z_{\min}, z_{\max})$, with x a cell in the SG and $z_{\min}$, $z_{\max}$ the lower and upper bounds of the simulated variable $Z(x)$. Three types of inequality constraints can be considered for $Z(x)$: (1) $z_{\min} \leqslant Z(x)$, if only $z_{\min}$ is defined, (2) $Z(x) \leqslant z_{\max}$ if only $z_{\max}$ is defined, and (3) $z_{\min} \leqslant Z(x) \leqslant z_{\max}$ if both $z_{\min}$ and $z_{\max}$ are defined. Note that this last expression can be used for any inequality data point by setting $z_{\min} = -\infty$ if the lower bound is undefined and $z_{\max} = +\infty$ if the upper bound is undefined.

Hence, a conditioning data set is split in two: a hard data set and an inequality data set.

2.2. Algorithm

The principle of the direct sampling algorithm (Mariethoz et al., 2010) is to successively populate the cells of the SG by pasting values from TI cells, while ensuring similar spatial configurations (patterns) in the vicinity of the simulated cell in the SG and of the selected cell in the TI. Three main parameters are needed: the maximal number N of cells in a pattern, the acceptance threshold t, and the maximal proportion f of the TI to scan. For the simulation of each SG cell, a random search is performed in the TI: as soon as the distance between the pattern around the simulated cell and the pattern around the scanned TI cell is less than or equal to t, the scan is interrupted and the value in the scanned TI cell is pasted in the SG. If a fraction f of the TI cells has been scanned, the search is interrupted anyway and the best scanned cell—the cell with the minimal pattern distance—is retained.

Whereas dealing with the hard data points is straightforward by pasting the given values in the SG in the beginning of the simulation, this approach cannot be applied for the inequality data points. To deal with such a situation, an idea is to modify the distance used to compare the patterns.

Let $d(x)$ be a pattern in the SG centered at the cell x. It is defined by N cells, $x_1, ..., x_N$, around x and information attached at each x_i, which can be either a known value $Z(x_i)$ or an inequality constraint given by $z_{\min}(x_i)$ and/or $z_{\max}(x_i)$. On the TI side, any considered pattern will have only known values. Let $d(y)$ be a pattern

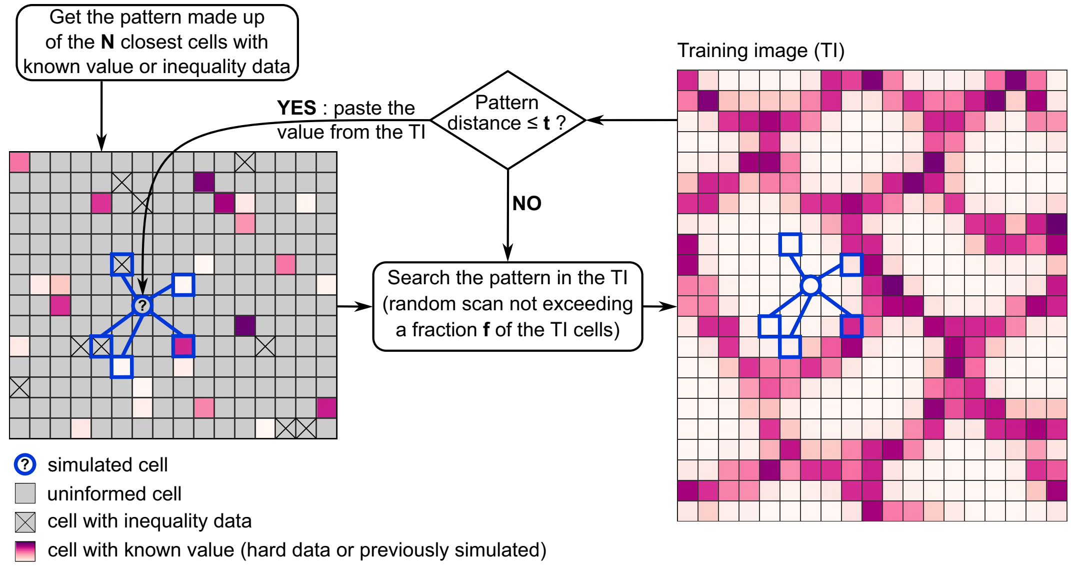
Get the pattern made up of the **N** closest cells with known value or inequality data
YES : paste the value from the TI
Pattern distance ≤ **t** ?
Training image (TI)
NO
Search the pattern in the TI (random scan not exceeding a fraction **f** of the TI cells)
?
? simulated cell
uninformed cell
cell with inequality data
cell with known value (hard data or previously simulated)

Figure 1. Illustration of the DeeSse algorithm with inequality data.

centered at y in the TI, with the value $Z(y_i)$ at the cell y_i, located by the lag vector $h_i = y_i - y = x_i - x$, for $i = 1, ..., N$. The distance between $d(x)$ an $d(y)$ is defined as follows,

$$D(d(x), d(y)) = \frac{1}{N}\sum_{i=1}^{N}\delta_i, \quad (1)$$

where

$$\delta_i = \begin{cases} \left|Z(x_i) - Z(y_i)\right| & \text{if the value is known at } x_i, \\ \max\left(z_{min}(x_i) - Z(y_i), 0\right) & \text{if only the lower bound is known at } x_i, \\ \max\left(Z(y_i) - z_{max}(x_i), 0\right) & \text{if only the upper bound is known at } x_i, \\ \max\left\{\max\left(z_{min}(x_i) - Z(y_i), 0\right), \max\left(Z(y_i) - z_{max}(x_i), 0\right)\right\} & \\ \qquad \text{if both the lower and upper bounds are known at } x_i. & \end{cases} \quad (2)$$

Note that the latter expression can be used for any inequality constraint with $z_{\text{min}} = -\infty$ (resp. $z_{\text{max}} = +\infty$) if the lower (resp. upper) bound is undefined, as mentioned in the previous section. Note also that even usual hard data can be handled by the last expression with $z_{\text{min}} = z_{\text{max}} = Z(x_i)$, the known value.

The proposed MPS method following the direct sampling strategy and accounting for inequality data is implemented in the DeeSse algorithm which is illustrated in Figure 1 and whose main steps are given by:

Input: SG, conditioning data set, TI, parameters N, t, f.

(1) Assign every hard data point in the SG.
(2) Localize every inequality data point in the SG.
(3) Define a random path visiting the cells in the SG without known value (cells without any information or with an inequality constraint).
(4) For each cell x in the path, do:
    (a) Retrieve the pattern $d(x)$ in the SG around x. It is composed of a maximal of N cells with a known value or corresponding to an inequality data point. The cells the closer to x (and possibly limited in a search ellipsoid) are retained.
    (b) Set $D_{\text{best}} = +\infty$ and y_{best} to "undefined," and scan the TI grid cells y randomly:

(i) If x is an inequality data cell and $Z(y)$ does not satisfy the corresponding inequality constraint, then skip the sampled cell y, that is, go to (iv). Otherwise, retrieve the pattern $d(y)$ centered in y in the TI (and whose the geometry is the same as that of the pattern $d(x)$), and compute the distance $D(y) = D(d(x), d(y))$ according to Equations 1 and 2.
(ii) If $D(y) < D_{\text{best}}$, then set $y_{\text{best}} = y$ and $D_{\text{best}} = D(y)$.
(iii) If $D_{\text{best}} \leqslant t$, then exit the (i-iv)-loop.
(iv) If a fraction f of the TI has been scanned, then exit the (i-iv)-loop.

(c) If y_{best} is still undefined, due to the inequality constraint at x, then choose a random cell y_{best} in the TI that verifies the constraint.
(d) Assign the value $Z(y_{\text{best}})$ to $Z(x)$.

Note that any hard data value z and any inequality value z_{min} and z_{max} are assumed to be compatible with the TI. Denoting by $Z_{min}^{(TI)}$ and $Z_{max}^{(TI)}$ respectively the minimal and maximal values found in the TI, this means that $Z_{min}^{(TI)} \leqslant z \leqslant Z_{max}^{(TI)}$, and $z_{min} \leqslant Z_{max}^{(TI)}$ and $Z_{min}^{(TI)} \leqslant z_{max}$.

Moreover, the considered variable Z is first normalized in the interval $[0, 1]$ before starting the simulation. The application $Z \mapsto (Z - Z_{min}^{(TI)}) \,/\, (Z_{max}^{(TI)} - Z_{min}^{(TI)})$ is applied to the TI values and to every data value (exact value, lower, and upper bound values). Then the variable is back-transformed at the end of the simulation. This allows to interpret the threshold parameter t as a tolerated error between 0 and 1, since the distance between two patterns will have a maximal value of 1.

2.3. Adapting the Method for Categorical Variable and Multi-Variate Case

An inequality data point for a categorical variable consists in a list of acceptable categories (facies) at a point (cell) in the SG. The approach developed above is straightforwardly adapted to that situation. The simulated variable Z is discrete, and the distance between patterns consists in computing the proportion of mismatching cells. Its expression is given by Equation 1, with δ_i equals to 0 or 1 defined as (adaptation of Equation 2): $\delta_i = 0 \Leftrightarrow Z(y_i) = Z(x_i)$ if the category $Z(x_i)$ is known at the cell x_i, and $\delta_i = 0 \Leftrightarrow Z(y_i) \in \mathcal{S}(x_i)$ if only the list $\mathcal{S}(x_i)$ of acceptable categories at x_i is known.

Inequality data points in the context of a categorical variable can be used to exclude some categories at some locations in the simulation domain.

The algorithm described in Section 2.2 is also easily extended to the multi-variate case. In this situation, several variables are simulated at each cell of the SG, using a TI with as many variables attached to its grid cells. The procedure for simulating one variable value (or the vector of values) in a SG cell consists in searching a location in the TI (by a random scan) such that all the pairs of patterns (one pair per variable) are sufficiently similar (in the SG and TI). Moreover, each variable Z_j is related to its own maximal number of patterns nodes N_j (and search ellipsoid) and its own acceptance threshold t_j, allowing to give more or less importance to the corresponding variable. The main changes in the algorithm of Section 2.2 are the following: in step (4), a pattern $d_j(x)$ is retrieved for each variable Z_j and the function $D(y)$ expressing the dissimilarity of the pattern located in y in the TI is replaced by the function $E(y) = \max_j \max\{(D_j(y) - t_j)/t_j, 0\}$ expressing the maximum over all the variables of a relative error with respect to the threshold, D_j being defined for the j-th variable as D. Details can be found in (Straubhaar et al., 2020). Following this strategy, the realizations will reproduce the spatial statistics within and between variables of the TI, while honoring usual and inequality conditioning data that can be considered for any variable.

2.4. Dealing With Dense Inequality Data Set

When a dense inequality data set is considered, following the algorithm above, and because the algorithm retains the N closest informed neighbors, the pattern retrieved from the SG early in the simulation process will be very compact, centered around the unknown location and most of its cells will correspond to inequality data. This prevents a proper reproduction of the spatial statistics of the TI at large scale. Adapting the way of retrieving the pattern around the simulated cell allows to avoid this drawback.

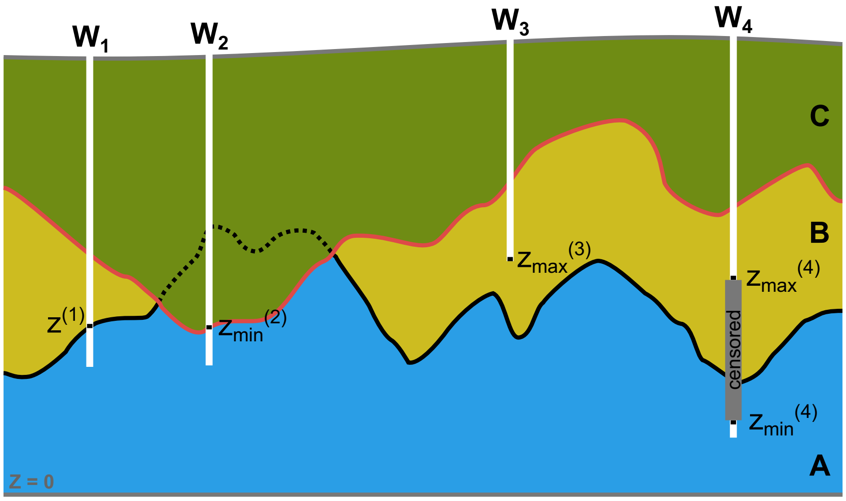

Figure 2. Illustration of different conditioning data types for the elevation Z of the interface between unit A and unit B (black curve) provided by four wells: hard data value $z^{(1)}$ for well W_1, lower bound $z_{min}^{(2)}$ only for W_2, upper bound $z_{max}^{(3)}$ only for W_3, and both lower bound $z_{min}^{(4)}$ and upper bound $z_{max}^{(4)}$ for W_4.

An additional parameter, $p_{\text{ineq}} \in [0, 1]$, is used to control the proportion of cells in the pattern provided by inequality data points. The retrieval of the pattern around the simulated cell x (step (4a) in the above algorithm) is adapted as follows. The SG cells around x are visited in an order such that their distance to x is increasing. As soon as a cell has a known value or is an inequality data point, it is retrieved and appended in a vector. When this vector reaches its maximal size N, the proportion of cells corresponding to inequality data points is computed. As long as this proportion exceeds the given parameter p_{ineq}, a further cell with a known value is searched around x and, if found, it replaces the last cell of the pattern provided by an inequality data point. Note that early in the simulation, a pattern may have more cells without fixed value than what is prescribed by the parameter p_{ineq}, because the amount of hard data points can be low.

An inequality data point $(x, z_{\text{min}}, z_{\text{max}})$ defines a target interval $[z_{\text{min}}, z_{\text{max}}]$ (with possibly $z_{\text{min}} = -\infty$ or $z_{\text{max}} = +\infty$) for the simulated value $Z(x)$. As this latter is determined by sampling the TI, the final target interval is $I(x) = [z_{min}, z_{max}] \cap [Z_{min}^{(TI)}, Z_{max}^{(TI)}]$. Therefore, the strength of the constraint depends on the length of the interval $I(x)$ (the larger the interval, the weaker the constraint) or more precisely on the proportion of the TI values falling in $I(x)$, and also on its relations with neighboring constraints. For example, two inequality constraints in neighboring points with a wide range but only slightly overlapping imply strong constraints. On the other hand, the distance from an inequality data point to the central pattern node (the simulated one) plays a key role: ignoring a data point close to the simulated cell may imply a loss of continuity in the resulting structures at small scale. The adopted strategy consists in ignoring some inequality data points to keep the pattern small while capturing the large scale structure. The criterion used to determine which inequality data points are dropped is based on the distance to the simulated cell (the farthest ones are replaced by hard data points) to also ensure the continuity of the simulated structures at small scale. Note that other custom criteria could be developed, accounting for the target intervals I, their relations, the distribution of the TI values and the distances to the central pattern node.

A brief sensitivity analysis to the parameter p_{ineq} is proposed in Section 3.3.

3. Applications and Examples

To illustrate the proposed methodology, we consider the simulation of elevation, for example, topography or interface between geological units. But first, we discuss how different types of conditioning data may arise in this situation.

Figure 2 shows a vertical cross section of a simple geological model with three stratigraphic layers, named A, B, C, from the most ancient to the most recent one. Assume that the goal is to simulate the elevation (Z)

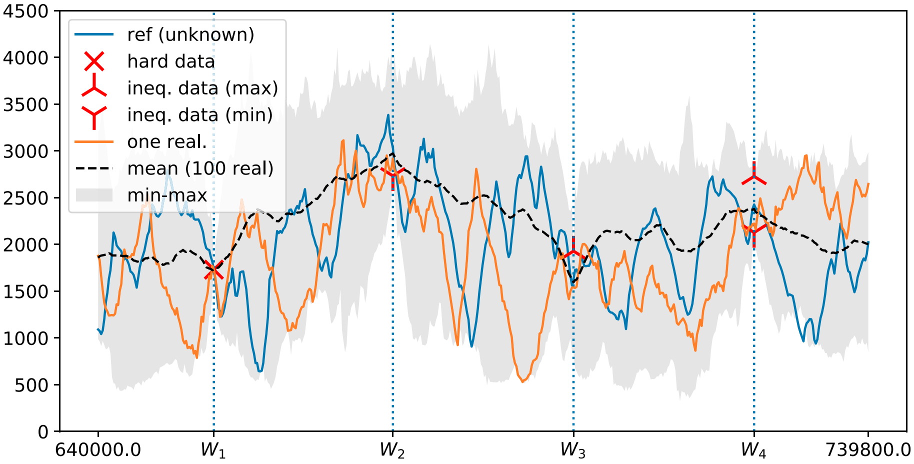

Figure 3. Results of 100 (uni-dimensional) realizations of profile of altitudes, with four conditioning data points in W_1, to W_4 of same type as in Figure 2 (unit for both axis: meter).

of the interface between the unit A and the unit B, depicted with a black curve in the figure. The well W_1 crosses the interface between A and B, and then a value $z^{(1)}$ of the elevation is obtained at the location of this well (hard data value). The three other wells provide inequality data points. Below the well W_2, the unit B has been eroded and is not present. As a consequence, the elevation of the top of unit A in W_2 is interpreted as a minimal value $z_{min}^{(2)}$ for the elevation at the location of this well. The well W_3 is not sufficiently deep to reach the unit A, then the elevation of the bottom of this well is interpreted as a maximal value $z_{max}^{(3)}$. Finally, imagine that measurements along a portion of the well W_4 is censored because a part of the core data have been lost or simply not recorded. The length of such intervals can be from a few decimeters to several hundreds of meters for deep boreholes. The bottom of the censored part belongs to the unit A and its top to the unit B, then the corresponding elevations provide a minimal value $z_{min}^{(4)}$ and a maximal value $z_{max}^{(4)}$ for the elevation of the interface.

In the next sections, the proposed method is illustrated on uni-dimensional and bi-dimensional examples where the elevation is simulated using parts of a digital elevation model of Switzerland with a resolution of 200 × 200 m (Federal Office of Topography [swisstopo], 2010) as TI.

3.1. Simulating a 1D Elevation Profile

As a first illustration of the proposed method, simulations of one dimensional profiles of altitudes are performed.

An elevation profile long 100 km from West to East in the central part of Switzerland is taken as a reference profile (blue curve in Figure 3), with one elevation height value every 200 m (500 points). Four equidistant points are selected along this curve and used to form a conditioning data set of the same type as shown in Figure 2: the first point, located at W_1, is a hard data point, the other ones are inequality data points with only a minimal value for W_2, only a maximal value for W_3, and both minimal and maximal values for W_4. The values in equality constraints are set to the reference value ± 300 m.

The TI (not shown) is composed of 380 profiles of 1,263 points (every 200 m, i.e., same resolution) along West-East direction. Note that the reference profile, as well as the 50 profiles in North and South of it, are not present in the TI. The main parameters for DeeSse were $N = 24$ (pattern extent is not limited by a search radius), $t = 0.01$, $f = 0.25$, and $p_{\text{ineq}} = 1$. The reference curve, one realization, the mean over 100 realizations and the range of values that they covered are shown in Figure 3. The curve of the realization depicts similar structures as the reference one, and conditioning data is honored.

In this example, as only 3 inequality data points are considered (at W_2, W_3, W_4), a rejection sampling approach can be applied. This method is numerically inefficient but it is known to be unbiased and we use it

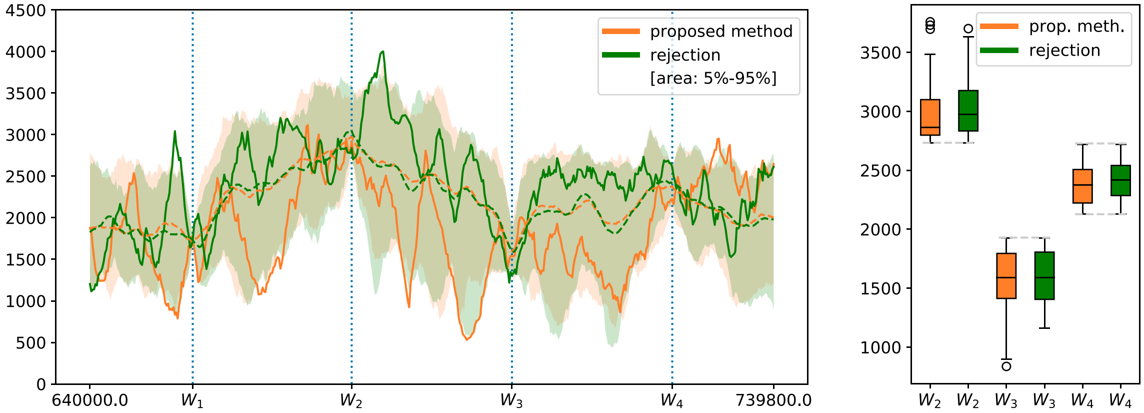

Figure 4. Comparison of the proposed method and the rejection approach; (left) one realization (solid line), mean (dashed line), and 5%–95% quantile area over 100 realizations for both methods; (right) distribution of the simulated elevations at inequality data locations for both methods (the dashed lines show the imposed bounds).

to check if our method provides similar results. The rejection sampling consists in generating a realization conditioned to the hard data points (only W_1 in this example), and accepting it if all inequality constraints are honored (or rejecting it otherwise). The procedure is repeated until the desired number of accepted realizations are reached. For this example, a total of 9,781 simulations have been done to obtain 100 realizations satisfying all the constraints. Figure 4 shows that the rejection and the proposed method give similar results for the structure of a single realization, as well as for the mean, the 5%–95% quantile area and the distribution of simulated values at inequality data points for the ensemble of the realizations. Therefore, the proposed method did not induce biases in this case, and allowed to save much computational time (rate of acceptation of ~1% for rejection). Finally, note that the rejection method will not be applicable any more with dense data set, contrary to the proposed approach that integrates inequality constraints during the simulation.

3.2. Simulating a Topographic Map

In this example, the goal is to reconstruct the incomplete image of the topography in the central part of Switzerland displayed in Figure 5 (from the digital elevation model of Switzerland with a resolution of 200 × 200 m, Federal Office of Topography [swisstopo], 2010). The red square drawn with a dashed line

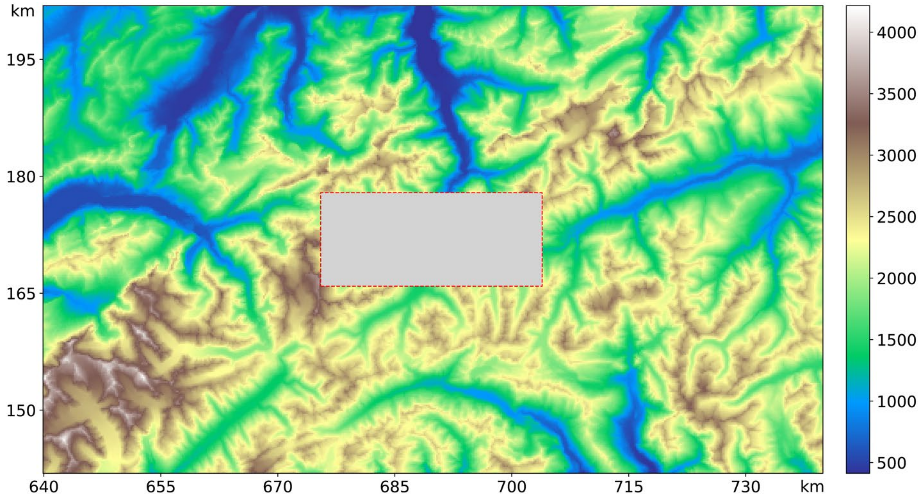

Figure 5. Input image (TI and usual hard data set) for 2D reconstruction displaying the topography in the center of Switzerland, with missing value in gray (unit for both axes: kilometer, unit for the color scale: meter).

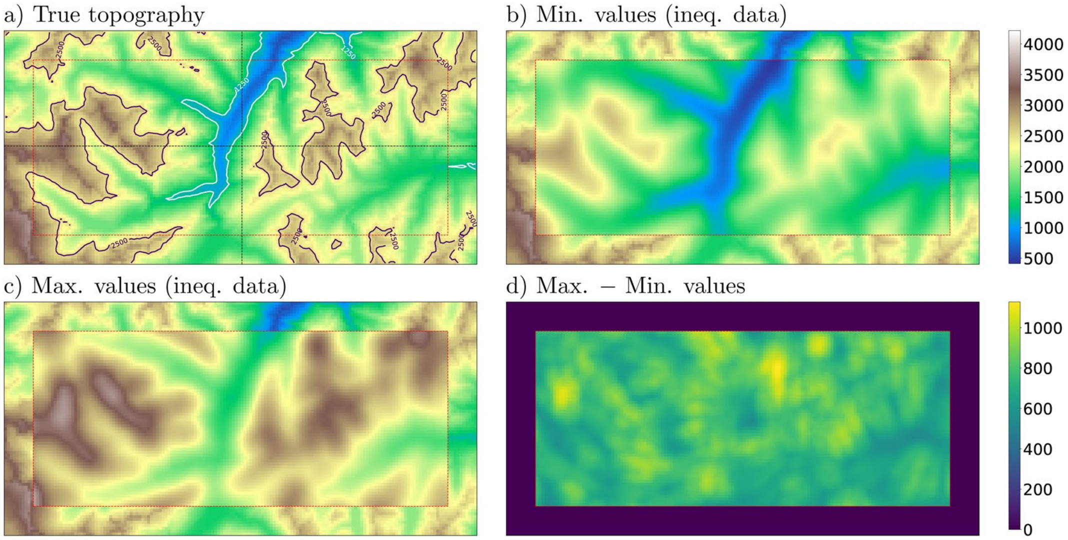

Figure 6. Maps defined on the missing part of Figure 5 (within the red square in dashed line): (a) the true topography; (b and c) inequality data maps: lower bound (b) and upper bound (c); (d) difference: map (c)–map (b). (Note: upper color scale is used for a, b and c.).

delineates the missing area (142 × 60 cells): the values (in gray) have been removed for the example. The incomplete image (whole domain, 500 × 300 cells) serves as the TI and as the usual hard data set for the next DeeSse simulations.

The map of the true topography that we aim at reconstructing is shown in Figure 6a, with contour lines for the altitudes 1,250 and 2,500 m. The dashed straight lines in black are the traces of the profiles shown further. Whereas for the further simulations, the true topography is ignored in the red square with dashed line, it is assumed that lower and upper bounds for the altitude are available on the missing part (Figures 6b and 6c). The gap between the lower and upper bounds is not uniform over the area (Figure 6d). Note that the two maps giving the bounds for the altitude have been constructed for this example by starting with the true topography and by applying several moving averages and shifting. For a better comparison, the same

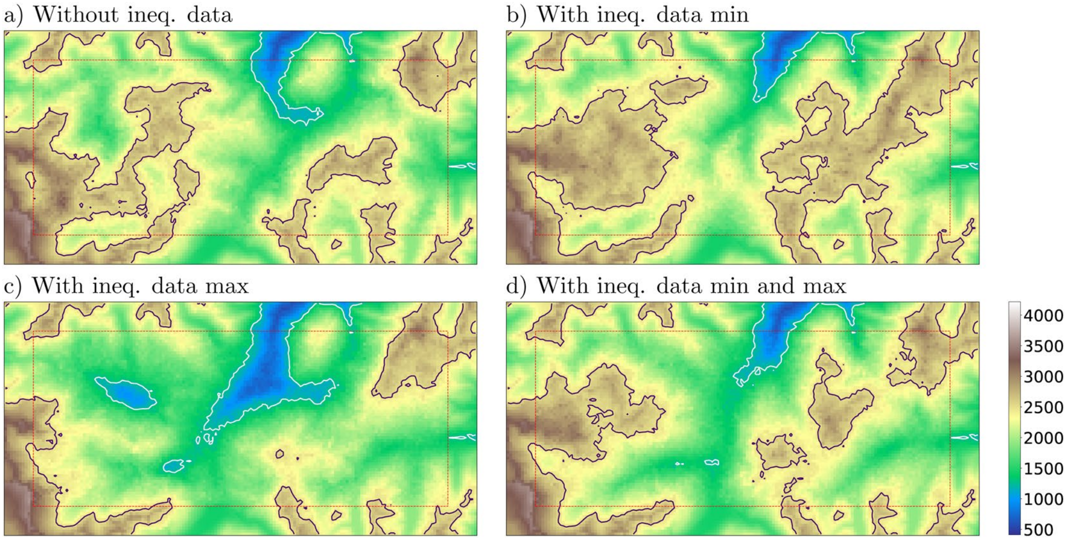

Figure 7. Examples of single realizations of topography using: (a) no inequality data, (b) only the minimal bound, (c) only the maximal bound, and (d) both minimal and maximal bounds.

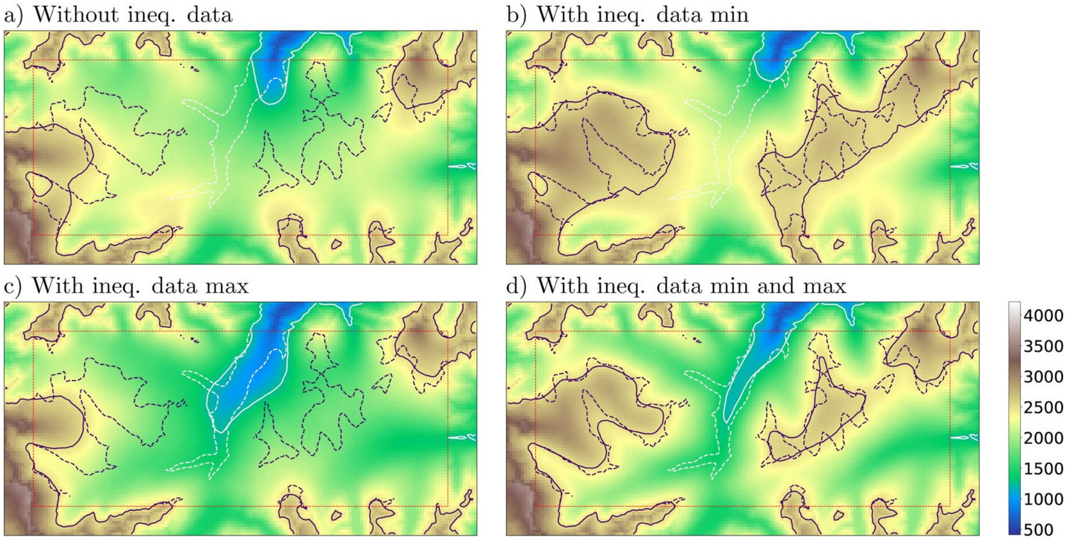

Figure 8. Mean (pixel-wised) over 100 realizations of topography using: (a) no inequality data, (b) only the minimal bound, (c) only the maximal bound, and (d) both minimal and maximal bounds.

color scale is used for the maps in Figures 5 and 6a–6c, and a band of 10 cells (pixels) around the red square is shown on the maps of Figure 6 for a better visualization.

Simulations have been performed for reconstructing the missing part of Figure 5, by considering four cases: (1) using no information on the missing part, (2) using only the map of minimal values (Figure 6b) as inequality data set on the missing part, (3) only the map of maximal values (Figure 6c), and (4) these both maps. For each case, 100 realizations have been generated with the following DeeSse parameters: $N = 30$ (number of pattern cells), $t = 0.01$ (acceptance threshold), and $f = 0.3$ (maximal scanned fraction of the TI in the known part). The parameter proportion of cells in the pattern provided by inequality data points is set to $p_{\text{ineq}} = 0.25$, which means that at most seven cells over the 30 ones in the pattern is provided by an inequality data point.

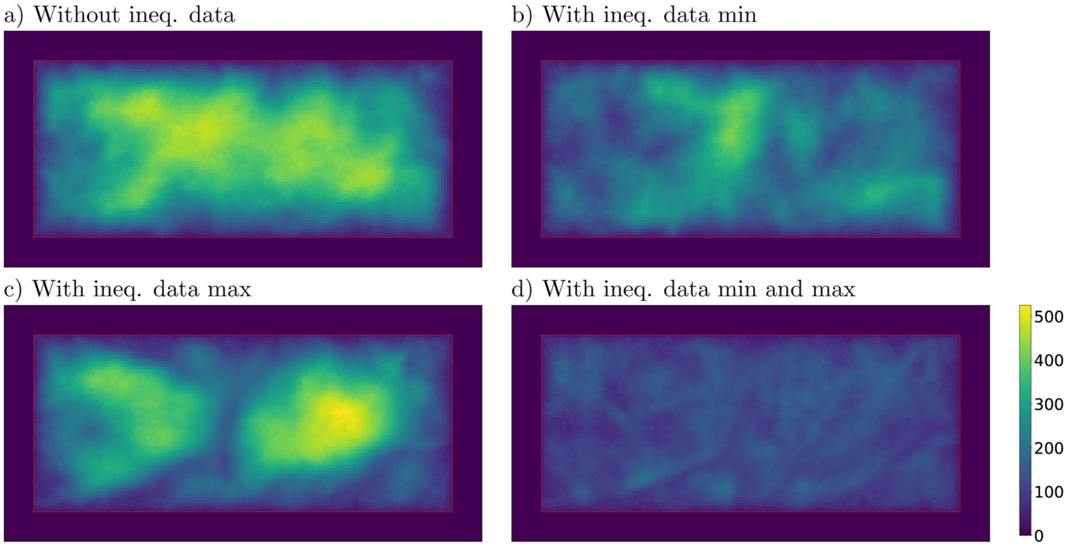

Figure 9. Standard deviation (pixel-wised) over 100 realizations of topography using: (a) no inequality data, (b) only the minimal bound, (c) only the maximal bound, and (d) both minimal and maximal bounds.

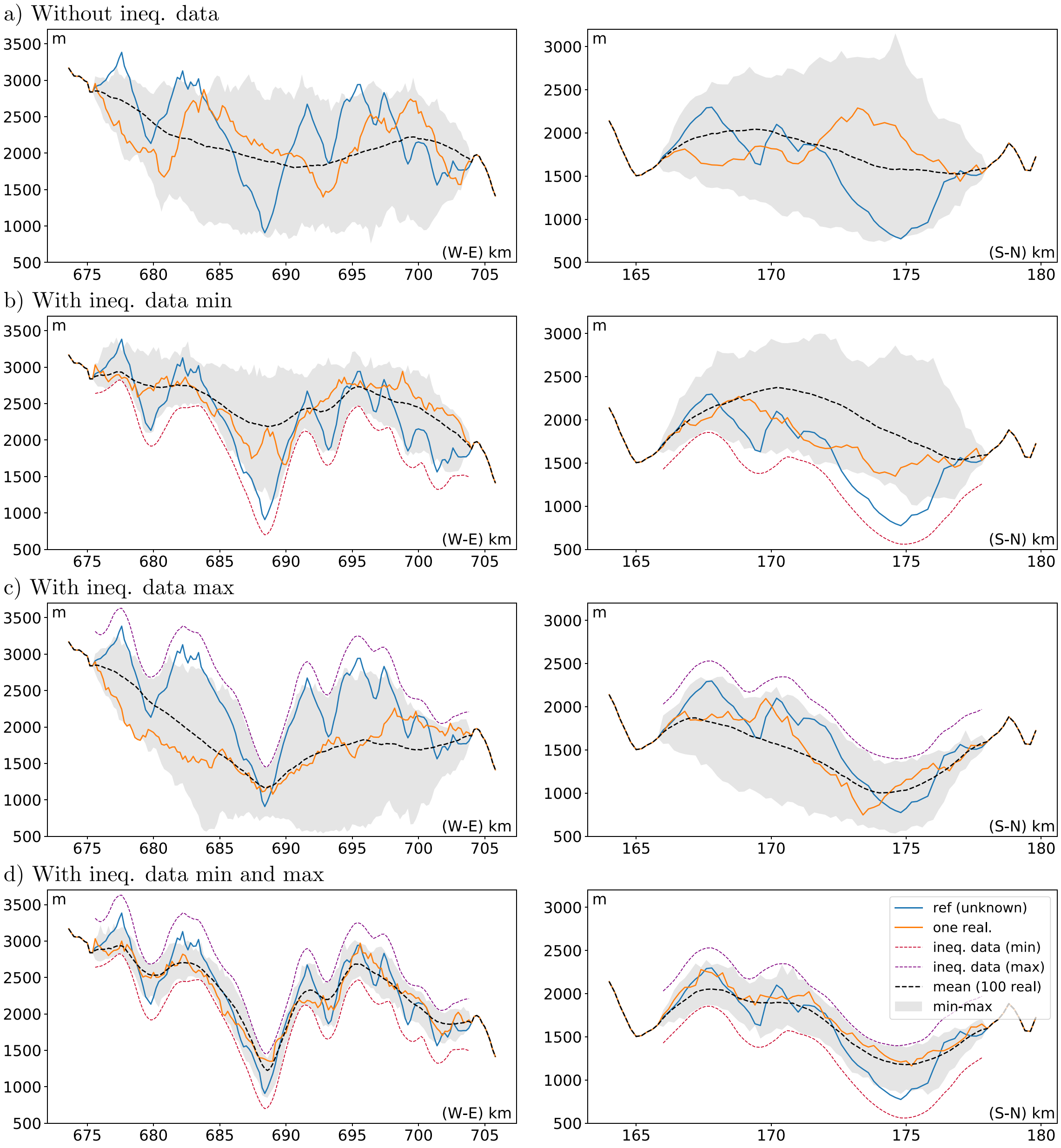

Figure 10. Profiles of altitudes along x-axis (left column) and y-axis (right column) taken from the four sets of 100 realizations of the previous bi-dimensional simulations. (Note: all the sections in the same column have identical scale on the vertical axis.).

One realization for each case is shown in Figure 7, the mean in Figure 8, and the standard deviation in Figure 9. For the single realizations and the mean maps, the contour lines for the same levels as in Figure 6a are shown, and the same color scale is used.

One observes on the single realizations (Figure 7) that the use of inequality data information does not affect the ability to reproduce the spatial features of the variable that can be found in the TI. Comparing the contour lines on the single realizations (Figure 7) and on the mean maps (Figure 8) one can see that using lower (resp. upper) bound values as inequality constraints helps locate the area with high (resp. low) altitudes in the simulation domain. However, using only minimal (resp. maximal) values as inequality data (Figures 7b and 8b, resp. Figures 7c and 8c) tends to underestimate the area of low (resp. high) altitudes. Nevertheless, if both minimal and maximal bounds are used, then low and high elevation areas are better

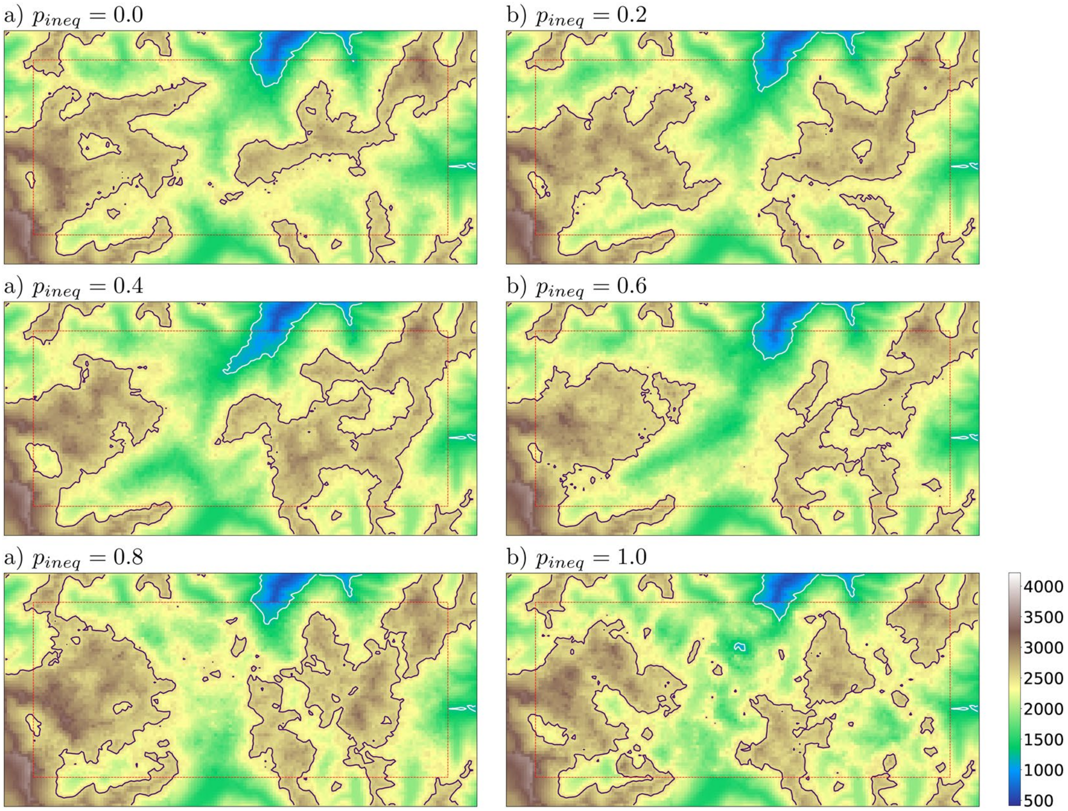

Figure 11. One realization of topography using only minimal values (Figure 6b) as inequality data, with different values for p_{ineq}.

located (Figures 7d and 8d), and the uncertainty over the set of realizations is reduced (compare the standard deviation maps in Figure 9).

Profile curves along *x*-axis (West to East) and along *y*-axis (South to North) going through the center of the reconstructed image (see their traces given in Figure 6a) are plotted on Figure 10 and corroborate the previous observations.

3.3. Controlling the Proportion of Inequality Data Cells in the Pattern

In this section, the sensitivity of the method to the parameter p_{ineq} is illustrated. This parameter controls the proportion of cells with inequality data in the pattern searched in the TI during the simulation process. Tests are performed using the same data set as in the previous section by varying p_{ineq} and using the same remaining DeeSse parameters. Ensembles of simulations are done with the six following values: $p_{\text{ineq}} = 0.0$, 0.2, 0.4, 0.6, 0.8, 1.0. As the number of cells in the pattern is at most $N = 30$, these values for p_{ineq} correspond respectively to at most 0, 6, 12, 18, 25, 30 cells with inequality constraints. Note that, even for $p_{\text{ineq}} = 0$, the algorithm ensures that the retained TI cell, from which the value is pasted into the simulation, honors the (possible) inequality constraint(s) locally at the simulated cell.

The situation where only the minimal values are considered as inequality data is considered for illustration. One realization and the mean (pixel-wise) over 100 realizations are shown for each value of p_{ineq} in Figures 11 and 12. One observes a slight deterioration of the spatial structures when p_{ineq} is increasing: the green and yellow areas in Figure 11 tend to lose their continuity. Indeed, as every simulated cell initially corresponds to an inequality data point, a high value for p_{ineq} implies that the 30 first (closest) cells around the simulated pixel will take a place in the pattern. As a consequence, the pattern will have a small lateral extension in the beginning of the simulation and the large scale structures cannot be captured well.

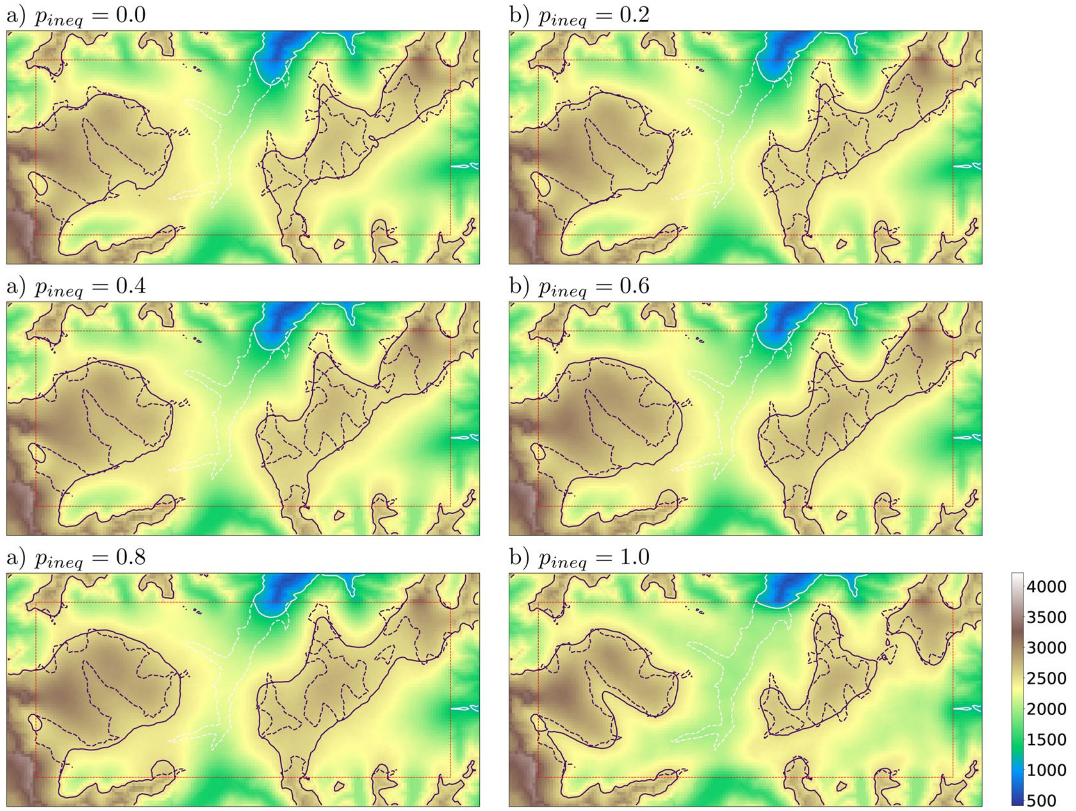

Figure 12. Mean over 100 realizations of topography using only minimal values (Figure 6b) as inequality data, with different values for p_{ineq}.

Moreover, whereas the high altitude area is better modeled in average, it becomes worse for the low altitude area (compare the contour lines in Figure 12). Therefore, the parameter p_{ineq} should not be too high to avoid undesired patterns.

Note that taking the maximal values only as inequality data leads to similar observations. Nevertheless, the differences are strongly attenuated when considering the minimal and maximal values as inequality data, because the simulations are highly constrained and very well guided. Finally, these experiments show that in practice a value of $p_{\text{ineq}}\approx 0.25$ (as used in the previous section) provides a good compromise for these examples where the inequality data set is very dense. Depending on the specific case, further trials would be required to adjust this parameter. In the case of sparse inequality data set a higher value of p_{ineq} would be fine.

4. Conclusions

The proposed method is an extension of the direct sampling MPS method allowing to handle inequality data, that is, inequality constraints on the simulated variable at some locations. It is implemented in the direct sampling algorithm DeeSse. The key point is an adaptation of the distance used to compare the pattern from the SG and the patterns scanned in the TI. Designed for a continuous variable, such constraints are taken into account by computing the gap between the value at a TI cell and the target interval (inequality data) at the compared cell in the SG and integrating a related contribution in the pattern distance. The method is straightforwardly adapted for the simulation of a categorical variable where an inequality constraint consists in specifying a list of acceptable categories.

Inequality constraints are common in real applications, and therefore the proposed approach meets a real need. The examples show that such constraints can be honored while keeping the ability of MPS technique to reproduce complex spatial features given in the TI. However, when a dense inequality data set is provided, the proportion of cells without fixed value in the pattern searched in the TI during the simulation process must be controlled to avoid a loss of quality in terms of the reproduction of desired spatial statistics. This is done with the help of a unique parameter giving the maximal proportion of such cells, which allows to obtain good results. Hence the proposed method remains easy to use.

Moreover, as this new capability requires only to adapt the pattern distance, it is compatible with most of options available in the DeeSse code, such that the use of orientation and/or scaling maps, target discretized distributions, or the use of multi-resolution TIs via Gaussian pyramids (Straubhaar et al., 2020). Whereas the examples presented in this study deal with a single variable in one or two dimension(s) (for a better focus on the new development), it is important to notice that it is also possible to handle inequality data in three-dimensional cases as well as joint simulations of multiple variables.

As a final remark, an alternative approach could be to use a Gibbs sampler (Freulon & de Fouquet, 1993) to simulate in a pre-processing step the values at the locations where inequality constraints are imposed, and in a second step simulating the rest of the locations. This could be an efficient alternative when the number of inequality data is small. However, for dense inequality data sets, the fact that the Gibbs sampler requires to run several iterations for each data, would most probably make the algorithm inefficient. The proposed approach has the advantage to be applicable and reasonably efficient both for dense and sparse inequality data sets.

Data Availability Statement

The data used in this study belong to the Swiss Federal Office of Topography and are freely available on their web site (Federal Office of Topography [swisstopo], 2010).

Acknowledgments
This work was supported by the Swiss National Science Foundation (Phenix project, grant number 200020_182600). We also thank the reviewers for their constructive remarks and suggestions.

References

Abrahamsen, P., & Benth, F. (2001). Kriging with inequality constraints. *Mathematical Geology*, *33*(6), 719–744. https://doi.org/10.1023/A:1011078716252

Chiles, J.-P., & Delfiner, P. (2012). Geostatistics: Modeling spatial uncertainty (2nd ed., Vol. 497): John Wiley & Sons.

Federal Office of Topography (swisstopo). (2010). *The digital height model of Switzerland with a 200m grid*. Retrieved from https://opendata.swiss/en/dataset/das-digitale-hohenmodell-der-schweiz-mit-einer-maschenweite-von-200-m

Freulon, X., & de Fouquet, C. (1993). Conditioning a Gaussian model with inequalities. In A. Soares (Ed.), *Geostatistics tróia'92* (pp. 201–212): Springer.

Linde, N., Renard, P., Mukerji, T., & Caers, J. (2015). Geological realism in hydrogeological and geophysical inverse modeling: A review. *Advances in Water Resources*, *86*, 86–101. https://doi.org/10.1016/j.advwatres.2015.09.019

Maatouk, H., & Bay, X. (2017). Gaussian process emulators for computer experiments with inequality constraints. *Mathematical Geosciences*, *49*(5), 557–582. https://doi.org/10.1007/s11004-017-9673-2

Mariethoz, G., & Caers, J. (2014). *Multiple-point geostatistics: Stochastic modeling with training images*. John Wiley & Sons.

Mariethoz, G., Renard, P., & Straubhaar, J. (2010). The direct sampling method to perform multiple-point geostatistical simulations. *Water Resources Research*, *46*. https://doi.org/10.1029/2008WR007621

Mariethoz, G., Straubhaar, J., Renard, P., Chugunova, T., & Biver, P. (2015). Constraining distance-based multipoint simulations to proportions and trends. *Environmental Modelling & Software*, *72*, 184–197. https://doi.org/10.1016/j.envsoft.2015.07.007

Michalak, A. (2008). A Gibbs sampler for inequality-constrained geostatistical interpolation and inverse modeling. *Water Resources Research*, *44*(9). https://doi.org/10.1029/2007WR006645

Robert, C. (1995). Simulation of truncated normal variables. *Statistics and Computing*, *5*(2), 121–125. https://doi.org/10.1007/BF00143942

Straubhaar, J. (2020). *DeeSse user's guide*. Neuchâtel, Switzerland: The Centre for Hydrogeology and Geothermics (CHYN), University of Neuchâtel.

Straubhaar, J., Renard, P., & Chugunova, T. (2020). Multiple-point statistics using multi-resolution images. *Stochastic Environmental Research and Risk Assessment*, *34*, 251–273. https://doi.org/10.1007/s00477-020-01770-8

Straubhaar, J., Walgenwitz, A., & Renard, P. (2013). Parallel multiple-point statistics algorithm based on list and tree structures. *Mathematical Geosciences*, *45*(2), 131–147. https://doi.org/10.1007/s11004-012-9437-y

Strebelle, S. (2002). Conditional simulation of complex geological structures using multiple-point statistics. *Mathematical Geology*, *34*(1), 1–21. https://doi.org/10.1023/A:1014009426274