# Data-Spatial Layouts for Grid Maps

**Nathan van Beusekom**
TU Eindhoven, The Netherlands

**Wouter Meulemans**
TU Eindhoven, The Netherlands

**Bettina Speckmann**
TU Eindhoven, The Netherlands

**Jo Wood**
City, University of London, UK

## Abstract

Grid maps are a well-known technique to visualize data associated with spatial regions. A grid map assigns each region to a tile in a grid (often orthogonal or hexagonal) and then represents the associated data values within this tile. Good grid maps represent the underlying geographic space well: regions that are geographically close are close in the grid map and vice versa.

Though Tobler's law suggests that spatial proximity relates to data similarity, local variations may obscure clusters and patterns in the data. For example, there are often clear differences between urban centers and adjacent rural areas with respect to socio-economic indicators. To get a better view of the data distribution, we propose grid-map layouts that take data values into account and place regions with similar data into close proximity. In the limit, such a data layout is essentially a chart and loses all spatial meaning.

We present an algorithm to create hybrid layouts, allowing for trade-offs between data values and geographic space when assigning regions to tiles. Our algorithm also handles hierarchical grid maps and allows us to focus either on data or on geographic space on different levels of the hierarchy. Leveraging our algorithm we explore the design space of (hierarchical) grid maps with a hybrid layout and their semantics.

**2012 ACM Subject Classification** Theory of computation → Computational geometry

**Keywords and phrases** Grid map, algorithms, trade-offs

**Digital Object Identifier** 10.4230/LIPIcs.GIScience.2023.10

**Supplementary Material** *Software (Source Code)*: `https://github.com/nvbeusekom/dataspatialhybridgridmaps`; archived at `swh:1:dir:49956cc7368207673acba37bc42be357ef4625f1`

**Acknowledgements** We want to thank Kevin Verbeek for useful discussions on the computational aspects of this paper.

## 1 Introduction

Many types of data have a spatial component: they relate to regions of interests, points of measurement, countries or other administrative zones, et cetera. Visualizing such data in a geographic map then allows studying patterns in the data that may be influenced by local or global geography – such as observing differences between rural and urban areas, or between northern and southern municipalities. However, as the data complexity increases, visualizing all data in a standard, geographically accurate map becomes infeasible, as precise geography may cause objects to become indistinguishably small or to clutter. A solution is to warp the geography instead, to create a better canvas for portraying the data, understanding that precise geography is not a necessity for observing higher-level patterns.

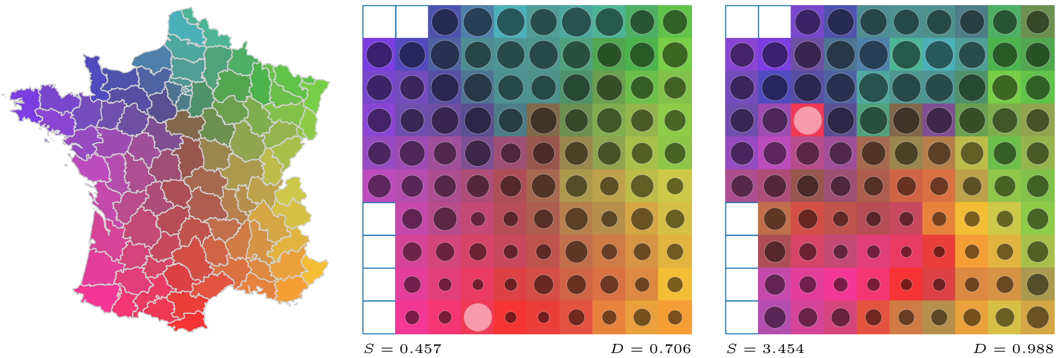



**Figure 1** Synthetic example of France and two grid map layouts. Color indicates the region, circle size indicates the data value. The modified value is highlighted with a light circle.

A popular example of such a warped geography are *grid maps* (also called tile maps), which are used, for example, by news outlets [1, 2, 6, 8, 14, 15, 21] and in the geo-visualization literature [10, 17, 19, 25, 28]. Grid maps schematize each region in the input into a simple shape (a *tile*), often a rectangle or hexagon, to then subsequently arrange these tiles into a grid, roughly according to geography. These tiles then act as a container for a visual encoding of the data associated with each region, which can take as simple a form as coloring, but might be as complex as charts for multivariate data. Effectively, a grid map is a spatially conditioned and arranged small-multiples visualization [22].

We call the assignment of regions to tiles in a grid a *layout*. The layout of a grid map is traditionally determined by the geography of the regions. There are many situations where one can expect that the geography and patterns in the data correlate, at least to some degree. Tobler, in his "first law of geography" [20], expressed this expectation as follows: "everything is related to everything else, but near things are more related than distant things". Local variations, however, may obscure interesting patterns in the data. For example, urban centers and adjacent rural areas frequently exhibit differences on socio-economic indicators, while cities, even when far removed from each other, often exhibit similar data values. Here, a purely spatial layout makes it difficult to obtain good overview of the data distribution. Given that a grid map already distorts space to some degree, one might hence consider layouts which distort space more to allow for a better representation of the data.

Consider the synthetic example of the departments of France in Figure 1. As data we are using the latitude of the region centroid (with some noise), with the exception of Ariège in the south, to which we assigned a northern value. In the grid maps color links cells to regions, and the circle size represents the data, that is, the latitude. The modified value is highlighted via a light circle. The middle of the figure shows a spatial assignment of departments to cells; regions are placed in a geographically coherent way. The right of the figure shows a *hybrid* layout that takes both space and data into account; regions are mostly placed to be geographically coherent, but also in such a way that similar values group together. The outlier can now easily be compared to regions with similar values, such that its place in the data distribution is clear. We posit that such hybrid layouts, that encode both data and space via location in the grid, might make it easier to observe patterns in the data that may otherwise stay hidden. Please note that, although we use color to indicate the effectiveness of our results, color is not required to indicate location in an eventual visualization and can hence be used to encode additional information.

**Results and organization.** In this paper we initiate the (algorithmic) study of hybrid layouts for grid maps. After covering some preliminaries in Section 2, we discuss in Section 3 measurements of how well space and data are represented in a given layout. Using our measures, in Section 4 we develop an algorithm that allows us to gradually transition from a spatial layout (which ignores data) to a data layout (which ignores space) and vice versa. Finally, in Section 5 we present a second algorithm which uses hierarchical properties of the geography to create hierarchical hybrids of spatial and data layouts. The resulting grid maps are visually pleasing and illustrate the potential of our techniques. We close in Section 6 with an extensive discussion of our results.

**Related work.** Algorithmically, grid maps were studied by Eppstein et al. [9], in the case that the number of tiles is roughly equal to the number of regions. The authors identified three main quality aspects: location (position in the grid with respect to the original geographic position in the map), adjacency (adjacent regions should map to adjacent tiles) and relative orientation (compass directions between regions). Eppstein et al. established that optimizing location is effectively a point-set matching problem, in which the sum of squared distances between the region centroids and their assigned tile centroids (the *displacement*) is to be minimized. Minimizing this displacement generally performs well also for the other criteria.

Meulemans et al. [11] broadened the perspective on grid maps by considering a suite of measures to assess layout quality in cases where the number of tiles well exceeds the number of regions. Here, empty tiles (gaps) can be used to give a more accurate reflection of the underlying geography. Generally speaking, a larger number of tiles allows for more accurate geographic representation at the expense of the size of the individual tiles, which in turn limits the complexity of the data visualization within each tile. The authors concluded that minimizing displacement still leads to good grid maps, but only in specific simple geographic settings. Subsequently, Meulemans et al. [12] described a pipeline to create high-quality grid maps for general geographic settings, by partitioning the input into simple pieces, leveraging cartogram techniques (see below) to create a tile configuration, and then minimizing displacement within each piece.

Grid maps have been applied hierarchically, for example, to create origin-destination maps (OD-maps) [18, 27]. OD-maps show the same (single) level both as the higher level structure, as well as the content of each tile, to visualize relational data between regions (such as migration). Here the hierarchical layouts are purely spatial, but they suggest nevertheless that nested layouts can be useful when investigating complex data.

Cartograms are another technique to overcome the constraints enforced by geographic detail. They deform the map such that every region has an area proportional to their data value. There are a wide variety of cartograms; most closely related to grid maps are contiguous cartograms that use schematic outlines, such as rectangular [4, 24], rectilinear [7], and mosaic cartograms [5]. The latter represent regions by multiple tiles in a grid, corresponding to data values; this is in contrast to grid maps which represent each region with a single tile. The quality of a cartogram is determined by two criteria that measure spatial coherence and data representation: (1) how well are the adjacencies and the relative positions of the geographic regions preserved in the cartogram, and (2) how large is the cartographic error (how well do the region sizes match their data values). Cartograms explicitly encode data, but similar data values cannot cluster unless the cluster is already present in the geography.

Treemaps show a hierarchy, using recursively partitioned rectangles, sized according to data values. Originally, there was no geographic space associated with the data, and optimization focused purely on achieving rectangles with low aspect ratios (i.e., close to squares); see for example [3]. Ordered treemaps [16] provide more control over the treemap structure, by relating it to a one-dimensional ordering of the data elements. Wood and Dykes [26] proposed to use two-dimensional (geographic) space to control the treemap structure, resulting in spatial treemaps – effectively a hybrid of cartograms and treemaps.

## 2 Preliminaries

Our input is a geographic map which consists of a set $\mathcal{R} = \{r_1, \dots, r_n\}$ of $n$ regions. Each region $r_i$ has a polygonal representation $p_i$, a centroid $c_i$ derived from it, and a data value $v_i$. Two parameters specify the target grid: its width $W$ (the number of columns) and its height $H$ (the number of rows). Together, they define a set $\mathcal{T} = \{t_{i,j} \mid 1 \leq i \leq W \text{ and } 1 \leq j \leq H\}$ of $W \cdot H$ *tiles*. Each tile is identified with its centroid. We focus on grid maps that use few tiles to represent the input regions, resulting in few gaps. That is, we set $H$ and $W$ such that $H \cdot W \geq n$, $(H-1)W < n$ and $H(W-1) < n$. Furthermore, we assume that the tiles in the grid and the geographic map have been aligned; see Eppstein et al. [9] for algorithms to optimize such alignment. We use square grids throughout our exposition, but our techniques readily generalize to other regular tile shapes.

The mapping of regions into a grid map is an injective function $L\colon \mathcal{R} \to \mathcal{T}$, the *layout*, which assigns each region to a unique tile in the grid. We use $\mathcal{A}_L$ to denote the set of all (unordered) pairs of regions that are assigned to adjacent tiles in layout $L$.

**Datasets.** To illustrate our techniques, we use three different geographic maps, and population data per region for each:

**FR:** The 94 departments of continental France. Source: `https://en.wikipedia.org/wiki/List_of_French_departments_by_population`

**EW:** The 331 Lower Tier Local Authorities of England and Wales, hierarchically aggregated into Wales and the 9 regions of England. Source: `https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/populationestimates/articles/demographyandmigrationdatacontent/2022-11-02`

**NL:** The 388 municipalities of the Netherlands in 2017, hierarchically aggregated into 12 provinces. Source: `https://opendata.cbs.nl/statline/#/CBS/nl/dataset/70072ned/table?dl=4675`

## 3 Layout quality

To understand and measure the quality of hybrid layouts, which take both space and data into account, we need to be able to measure how well space and data are represented in a given layout. In the following, we hence consider the two extremes: spatial layouts that ignore the data values and data layouts that ignore the geographic space.

### 3.1 Spatial layouts

A *spatial layout* is based purely on the geography of the input map and ignores the data values; in other words, it is a traditional grid map. We hence use the results of Eppstein [9] to compute spatial layouts by minimizing the total squared displacement between the region centroids and the centers of the grid tiles. This method crucially relies on the alignment of the geographic map and the grid: translation of the map can result in a different layout. As mentioned in Section 2, we assume that the input map and the grid have been aligned well, using known methods [9]. The result on the France dataset is shown in Figure 2. In the following we discuss how best to measure how "spatial" a given layout is.

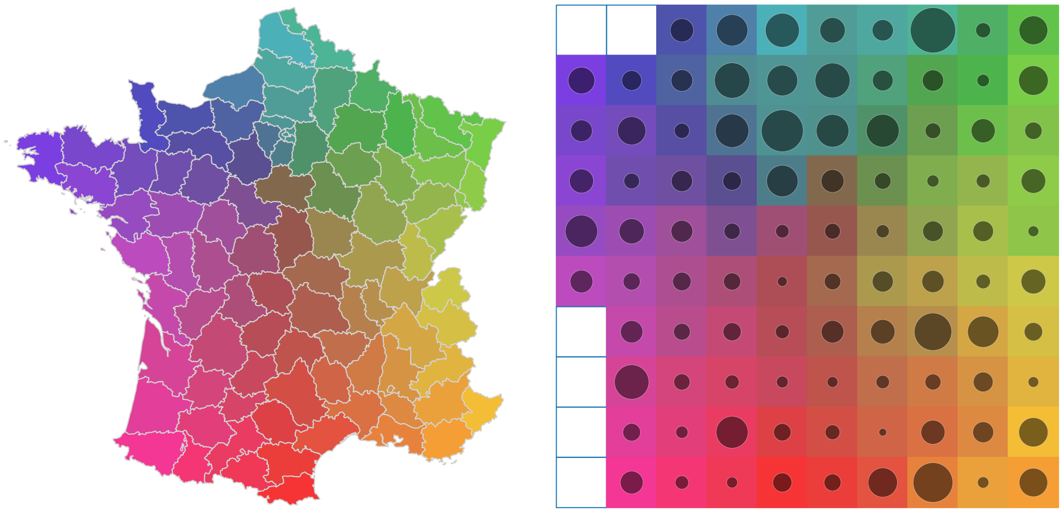



**Figure 2** The spatial layout for FR: color encodes location, circles sizes encode data values.

**Spatial distortion and spatial correlation.** We compute spatial layouts by minimizing displacement. Hence, a priori, displacement seems the logical choice to measure how well space is represented in a layout. However, displacement is inherently a global measure and as such not well suited for the local changes to the layout that occur during the kind of gradual transitions between spatial and data layouts that we envision for our hybrid layouts. *Spatial distortion* is a local measure for the distance and direction between regions that have been assigned to adjacent tiles. Specifically, with some normalization, we measure the spatial distortion $S$ of a layout $L$ as follows:

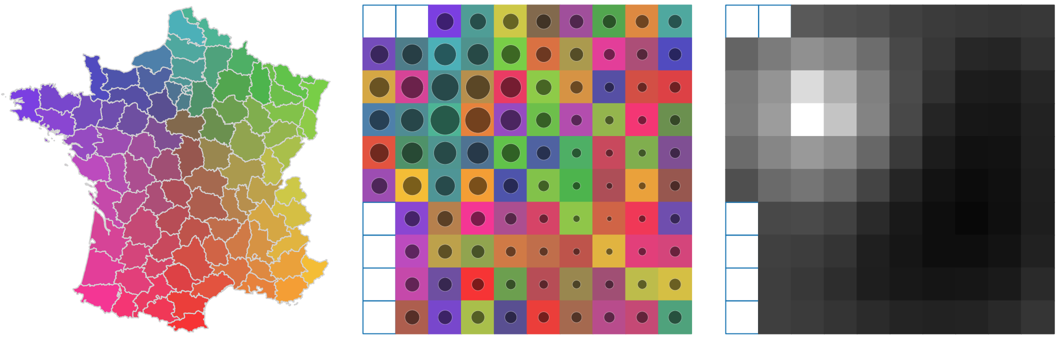
$$S(L) = \frac{1}{\mathcal{A}_L} \sum_{\{r_i, r_j\} \in \mathcal{A}_L} ((c_i - c_j) - (L(r_i) - L(r_j)))^2.$$

Here $c_x$ and $L(r_x)$ are vectors expressed in unit lengths (i.e. tile widths). That is, we average the difference of the vectors between two adjacent tile centers and their assigned region centers. We use vector subtraction such that direction is also taken into account. We say that a layout $L$ has high spatial correlation, when the spatial distortion $S(L)$ is low. Previous work [11] has established that minimizing displacement also generally results in low spatial distortion for spatial grid maps. We expect that minimizing $S(L)$ is NP-Hard: a reduction from Euclidean TSP may follow to minimizing $S(L)$ on a $n \times 1$ grid. Hence, our results are generally not (Pareto-)optimal.

In our figures, we color regions by their position in geographic space using a gradient. As a result, small geographic distances between adjacent tiles translate to low color variations – a layout that has high spatial correlation hence visually demonstrates smooth color changes.

## 3.2 Data layouts

A *data layout* is based purely on the data and ignores the geography of the input map. In contrast to a spatial layout – which corresponds to a traditional grid map – there is no one established way to create such layouts. We generally expect a data layout to cluster similar values together, especially at the high and low extremes of the value range. In a one-dimensional grid (that is, an array), this is readily achieved by sorting the regions by



$S = 20.897$ $D = 0.960$

**Figure 3** A data layout with a peak of high values, a valley of low values, and average values along the edges and through the diagonal.

value. However, in a two-dimensional grid, the optimum is less obvious. Before discussing our method to compute a data layout, we first define a measure for how well the data is represented in a given layout.

**Data correlation.** A grid map is by definition a map with well-defined edge adjacencies between tiles. Hence we can leverage spatial auto-correlation measures on the grid map space to measure similarities between data values of those regions that are mapped to tiles which are adjacent (or generally close) in the grid. Moran's $I$ [13] is a general measure for spatial auto-correlation which can be flexibly configured for various models of spatial proximity, via weights defined between each pair of regions. We want to focus on local relations and hence use a weight of 1 between adjacent tiles, and 0 between all other pairs. We defined the *data correlation* $D$ of a layout $L$ as the value of Moran's $I$ for a layout $L$:

$$D(L) = \frac{n}{|\mathcal{A}_L|} \frac{\sum_{\{r_i, r_j\} \in \mathcal{A}_L} (v_i - \bar{v})(v_j - \bar{v})}{\sum_{r_i \in \mathcal{R}} (v_i - \bar{v})^2}.$$

Here $\bar{v}$ denotes the average of all $v_i$. The value of $D$ is always between $-1$ and 1, where values towards 1 indicate a strong correlation of data values, and values towards $-1$ indicate a strong inverse correlation of data values.

Regions whose data values are significantly higher or lower than the average contribute most to the data correlation if they are placed next to other regions with the same deviation. That is, in a layout with high data correlation, very low and very high values cluster together, as we would expect from a data layout. Note that tiles along the boundary have fewer neighbors than interior tiles and hence have a smaller impact on the data correlation. This naturally attracts tiles with average data values to the boundary and onto a diagonal, separating a peak and a valley, see Figure 3.

**Computing layouts.** As mentioned above, computing an optimal data layout in one dimension is simply sorting. So one can wonder if there is a two-dimensional equivalent? A naive approach would fill row after row by the next highest value – that is, effectively sort in the same manner as squarified treemaps do [3]. In such a layout many tiles can be expected to not be adjacent to other tiles of similar values in more than two of the four directions and hence the data correlation is generally low. There are many other possibilities to map a one-dimensional order of the regions onto the grid along a space-filling curve. While some such mappings do better than others in terms of clustering similar data values, they all necessarily contain adjacent tiles with data values that are far removed from each other in the order. As a result, none of these sorted layouts optimize data correlation.

We hence computed (near-)optimal data layouts using simulated annealing, repeated for a set of random layouts. Effectively, we used our constrained annealing approach described in the next section, without constraints. We observed very slight variations in results due to randomization, but all with similar values for $D$. The layout with the highest data correlation, depicted in Figure 3, was used as the data layout in our experiments.

## 4 Hybrid layouts

In the previous section we introduced the two layouts that form the end of our spectrum: the spatial layout which ignores the data values and the data layout which ignores the geographic space. We also defined two measures to asses how "spatial" or how "data" a given layout is: the spatial distortion and the data correlation. In this section we now aim to compute meaningful *hybrid layouts* that represent both space and data to varying degrees.

Both measures are based on distances between tiles, but nevertheless, they do not live in the same mathematical space. It is hence unclear how to combine them in a meaningful way when computing and assessing hybrid layouts. We therefore take the following approach: we optimize for one measure while constraining the other. That is, we start on either end of the spectrum, constraining either space or data, and then optimize for the other, while slowly releasing the constraints. This creates two ranges of layouts, from space to data and vice versa, which allow us to explore the space of hybrid layouts and the corresponding trade-offs.

**Simulated annealing.** We use a constrained variant of simulated annealing: we define some slack parameter $\sigma$ and we allow the algorithm to search only the space of layouts that are at most $\sigma$ different from either the spatial or the data layout. Concretely, our algorithm is given an initial layout and performs random swaps of tiles. This follows standard simulated-annealing practice, with the addition that any swap that results in a layout exceeding the given slack $\sigma$ from the initial layout is always rejected. We use a starting temperature $T_s = \frac{\delta_s}{\ln(0.5)}$, an ending temperature $T_e = \frac{\delta_e}{\ln(10^{-9})}$, $10^7$ iterations, and exponential multiplicative cooling with factor $(T_e/T_s)^{10^{-7}}$. For improving the data correlation, we set $\delta_s = 10^{-3}$ and $\delta_s = 10^{-1}$. For reducing the spatial distortion we set $\delta_s = 10^{-2}$ and $\delta_s = 1$. Due to locality of our measures, each iteration takes $O(1)$ time.

**Spatial to data.** To explore the trade-off from the spatial extreme, we initialize the annealing algorithm with the spatial layout $L_\mathrm{s}$ and optimize for data correlation $D$, using a spatial slack parameter $\sigma_S$. Any swap that results in a layout $L$ with $S(L) > S(L_\mathrm{s}) + \sigma_S$ is rejected.

The results for France with increasing values of $\sigma_S$ are shown in Figure 4. We observe that even for a tiny amount of spatial slack $\sigma_S = 0.001$, the data correlation already increases considerably. Apart from the measured improvement, we indeed observe that similar values are grouped together: a peak of high values occurs in the northern blue area. However, we do not observe further significant improvement until $\sigma_S$ is increased to 1. For $\sigma_S = 1$ and $\sigma_S = 2$ two southern peaks emerge in the purple and orange area. These peaks are merged together into a single southern peak for $\sigma_S = 5$, yet still separated from the northern high values. At this point we also see the small values cluster in the center of the grid. Finally, the northern and southern peaks merge at $\sigma_S = 10$. This corresponds to an increase in data correlation $D$, which is already close to $D(L_\mathrm{d})$. At this point most of the spatial correlation is already lost. Increasing the spatial slack further leads to only minor improvement of $D$, while even further increasing the spatial distortion. We observe similar behavior in the results for EW (Figure 5) and for NL (6). However, the improvement of the data correlation happens more gradually. This might be due to the large number of regions, presenting the simulated annealing algorithm with more possibilities, while maintaining a similar spatial slack.

**Data to spatial.** To explore the trade-off from the data extreme, we use an analogous implementation of our constrained annealing algorithm. We initialize it with the data layout $L_\mathrm{d}$ and reduce the spatial distortion $S$, using a data slack parameter $\sigma_D$: any swap that results in a layout $L$ for which $D(L) < D(L_d) - \sigma_D$ is rejected.

Figure 7 shows our results for France with increasing values of $\sigma_D$. We again observe that allowing a small data slack $\sigma_D = 0.001$ already leads to a considerable reduction of spatial distortion. Visually, the data distribution is nearly identical to $L_\mathrm{d}$, yet regions are starting

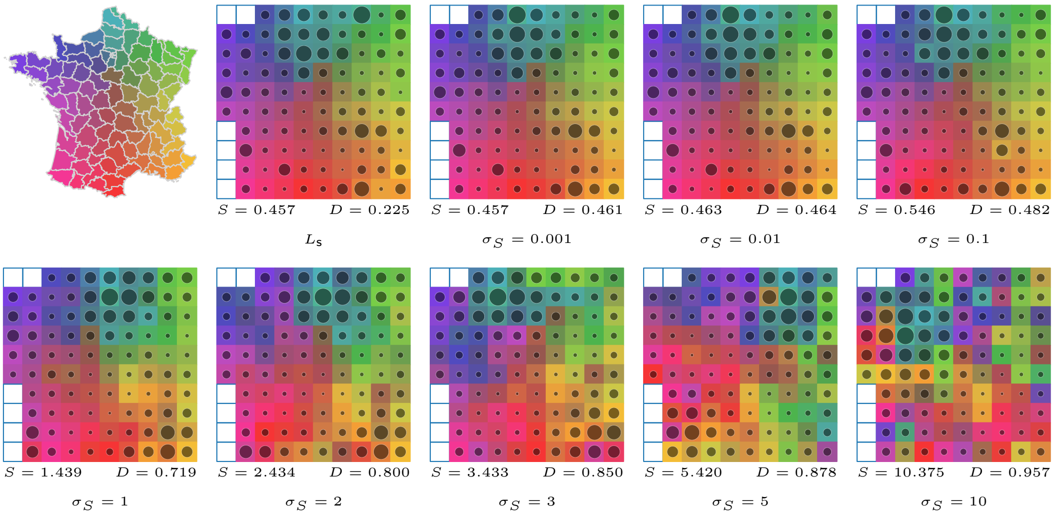



**Figure 4** Hybrid layouts for FR with increasing spatial slack $\sigma_S$.

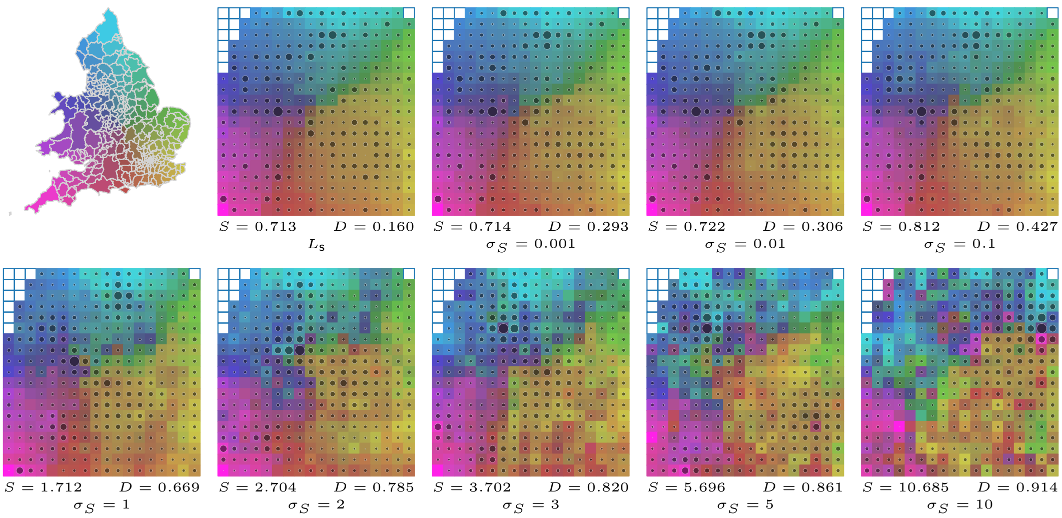



**Figure 5** Hybrid layouts for EW with increasing spatial slack $\sigma_S$.

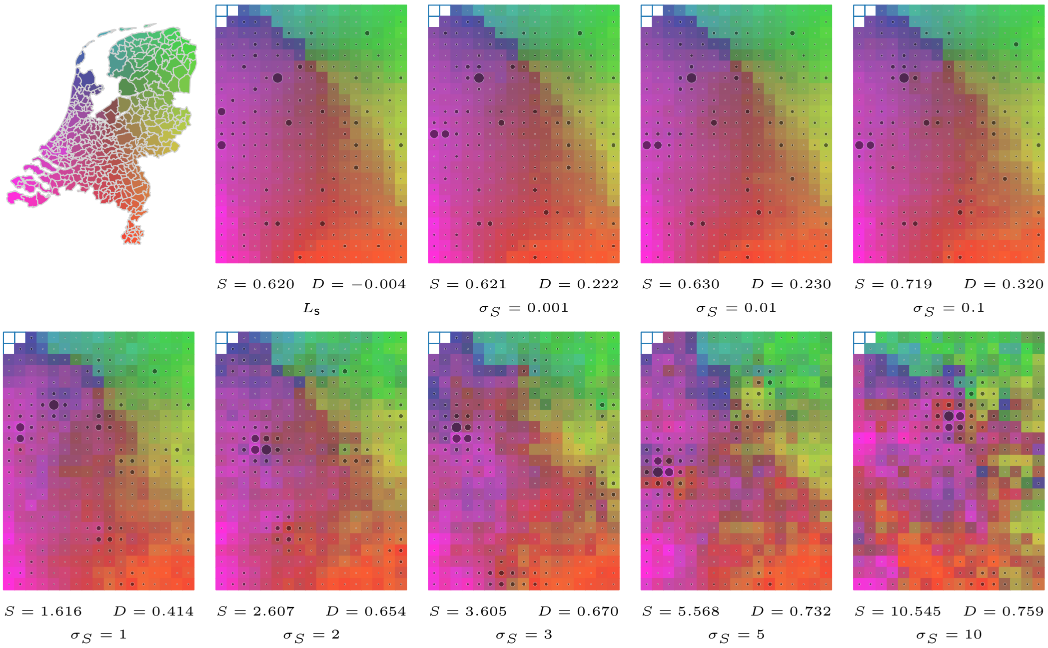



**Figure 6** Hybrid layouts for NL with increasing spatial slack $\sigma_S$.

to cluster if they are geographically nearby, indicated by smoother color changes. The effect further strengthens for $\sigma_D = 0.005$, where it becomes easier to locate most regions with low and average values from some particular areas. As $\sigma_D$ grows, more patches of similar colors appear, and less similar data values become adjacent. At $\sigma_D = 0.1$ the average values seem visually unorganized, and at $\sigma_D = 0.3$ some of the larger values are far from other peaks of large values. Most regions are now geographically grouped, apart from the orange regions. At $\sigma_D = 0.5$, most data correlation is lost, but the spatial distortion is close to $S(L_s)$. Again, the results for EW (Figure 8) and for NL (9) show similar behavior, though the improvement with low amounts of slack seems more significant. This might be due to

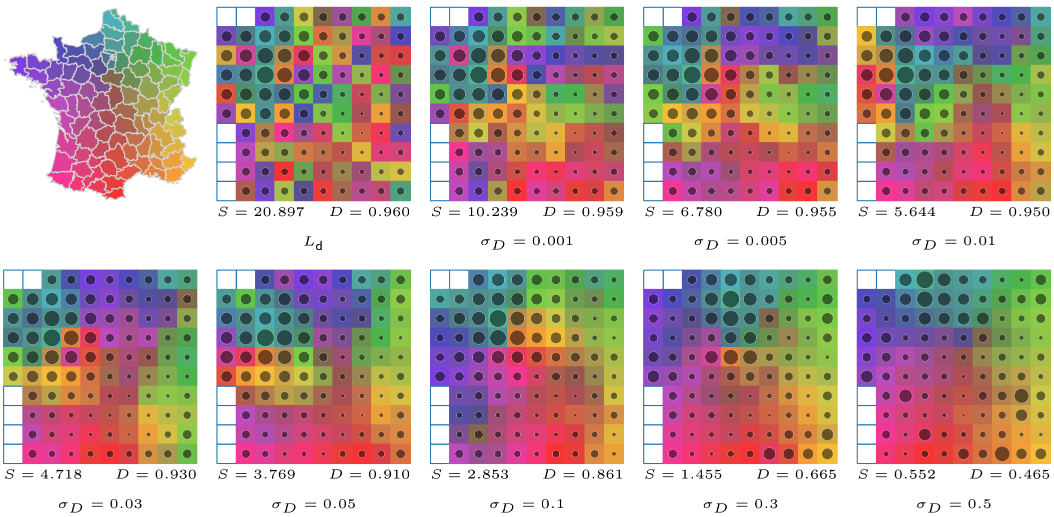



**Figure 7** Hybrid layouts for FR with increasing data slack $\sigma_D$.

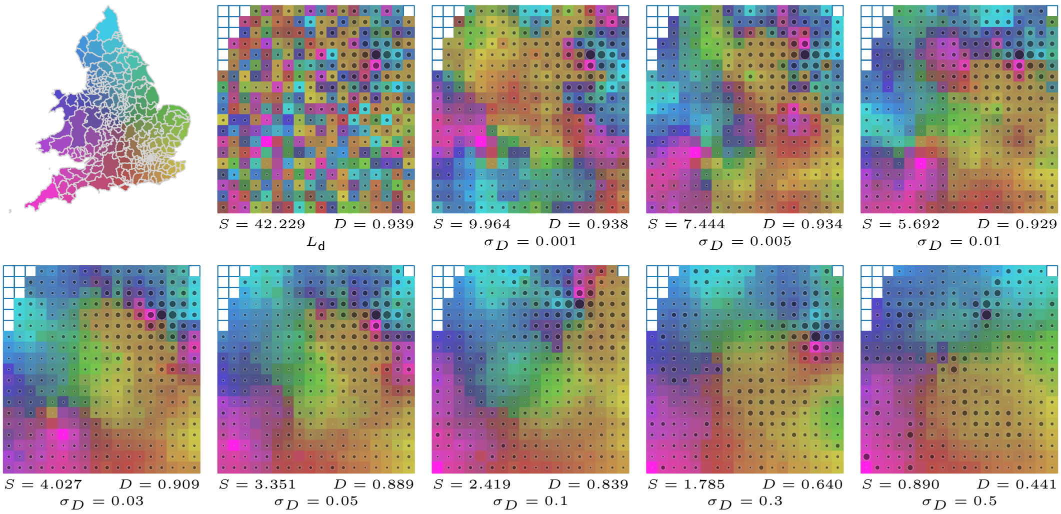



**Figure 8** Hybrid layouts for EW with increasing data slack $\sigma_D$.

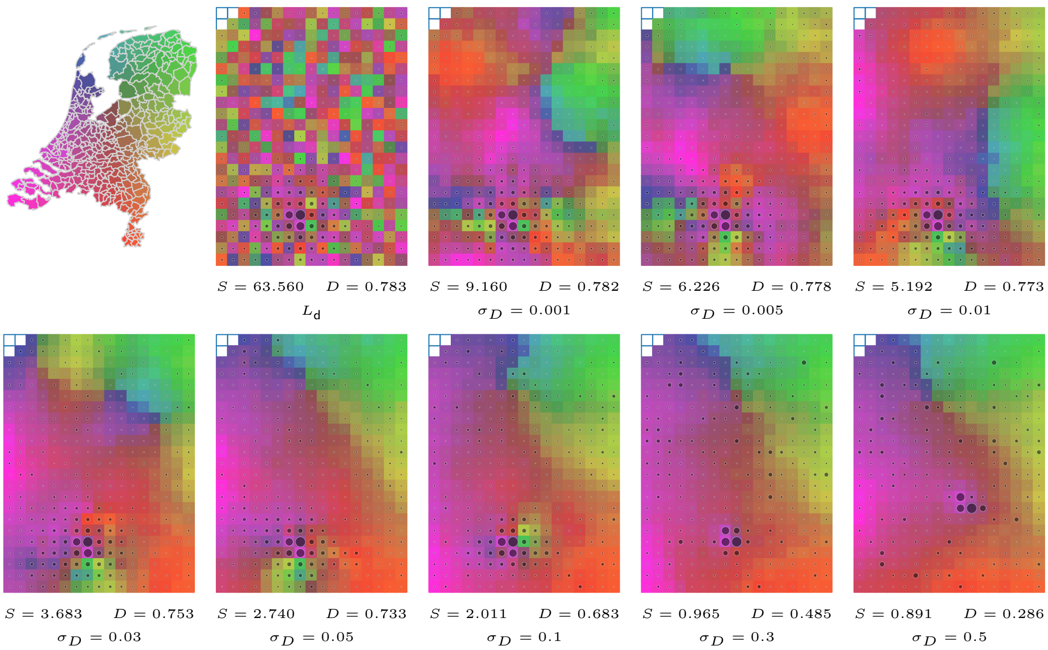



**Figure 9** Hybrid layouts for NL with increasing data slack $\sigma_D$.

the many similar data values in EW regions, giving the algorithm many valid swapping possibilities. We observe that it takes a large amount of slack before the outlying values move to a more suitable spatial position, with the largest value not even being in an optimal spatial position at $\sigma_D = 0.5$.

**Analysis.** There are layouts in both sequences that strictly outperform layouts in the other sequence. For example, the layout of France with $\sigma_D = 0.001$ has higher data correlation *and* lower spatial distortion than the layout with $\sigma_S = 10$. Similarly, the layout with $\sigma_S = 1$ outperforms the layout with $\sigma_D = 0.3$. We conclude that the trade-off can be efficaciously approached from both ends using our constrained simulated-annealing approach. Both methods readily achieve improvements using only small slack values. High slack values achieve reasonable results but are generally outperformed by the opposite approach with low slack values. We observe that the EW and NL layouts produced with high slack values are less close to the opposite extremes than is the case for France. They are hence still more of a hybrid layout. Increasing the slack further would eventually lead to the opposite extreme.

## 5 Hierarchical hybrid layouts

In the previous section we explored the trade-off between data and spatial layouts by creating sequences of layouts from both extremes. Each sequence focuses on one aspect, space or data, and integrates the other. While this approach arguably creates meaningful hybrid grid maps in the vicinity of the two ends of the spectrum, in the conceptual middle of the trade-off, the results might become difficult to interpret: for any given cluster of similar regions, it is unclear if this cluster was formed based on spatial or data correlation, which may hinder understanding of relations between clusters. Hence, in this section, we explore a more structured approach to integrate space and data. Specifically, we propose to use hierarchical information about the regions, effectively creating a *hierarchical grid map*. These zones may be administrative zones or deduced from the geography. It is less clear what meaningful hierarchies based on data would be; we hence focus on geographic hierarchies.

Within a zone regions may be arranged to create a spatial or data layout, while still displaying the zone affiliation. Meanwhile, the zones may also be arranged in a spatial or in a data layout, as to convey information about their relations on the higher level. For example, in a dataset with neighborhoods as regions and cities and rural areas for zones, the cities could re-arrange themselves to cluster together, separating themselves visually from rural zones. Yet, within each city (or within the rural areas), the spatial structure of neighborhoods is maintained. To achieve legibility across hierarchical level, it may be desirable to retain connectivity of regions within the same zone.

In the following we assume that our input map is augmented with a set $\mathcal{Z} = \{z_1, \ldots, z_k\}$ of $k$ *zones*. These zones partition $\mathcal{R}$, and we assume that they each capture a geographically somewhat coherent set of regions, such as provinces in a country. Furthermore, the data values of all regions in a zone $z_i$ are aggregated into a value $a_i$ for the zone – the form of this aggregation (e.g., average or sum) depends on the nature of the data and the desired effect in the map. We use administrative, established hierarchies in our experiments, aggregating based on the sum, as our data values represent population.

The central idea of our algorithm is then to leverage the hierarchy: we separate spatial and data aspects on different hierarchical levels, and finally blend the two together into a single layout. Each level of the hierarchy can independently be assigned to be a spatial layout, or a data layout. For simplicity, we assume a 2-level hierarchy. We hence select either data or spatial layouts on the zone-level and either data or spatial layouts on the region-level.

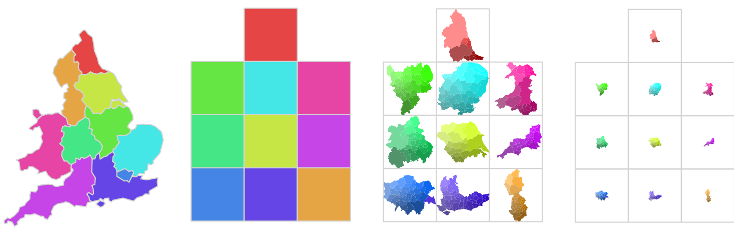
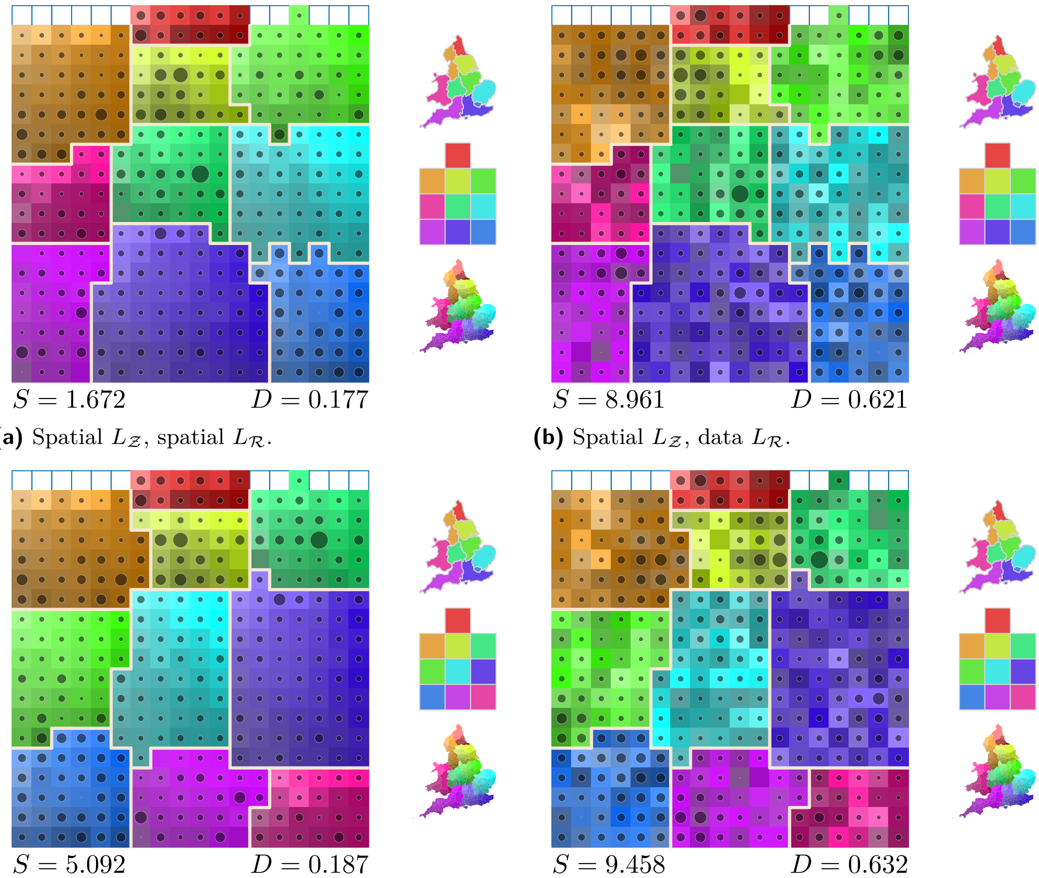


**Figure 10** Zone-level transformations: zones of EW; zone-level layout $L_{\mathcal{Z}}$ (data layout); regions of each zone fitted to the assigned tile; each zone scaled by factor $\lambda$, $\lambda = 0.25$ for illustration.




**(a)** Spatial $L_{\mathcal{Z}}$, spatial $L_{\mathcal{R}}$.

**(b)** Spatial $L_{\mathcal{Z}}$, data $L_{\mathcal{R}}$.

**(c)** Data $L_{\mathcal{Z}}$, spatial $L_{\mathcal{R}}$.

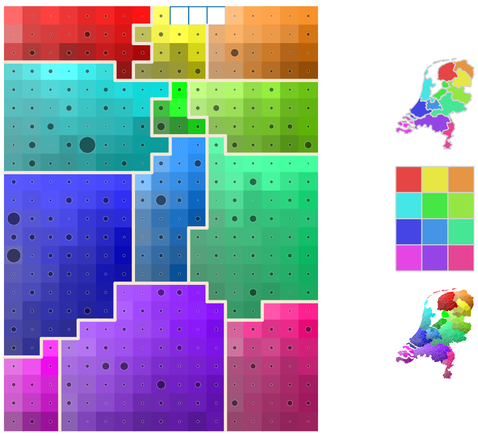
**(d)** Data $L_{\mathcal{Z}}$, data $L_{\mathcal{R}}$.

**Figure 11** Hierarchical grid map for EW. With each, the smaller maps represent from top to bottom: the zones $\mathcal{Z}$, the zone layout $L_{\mathcal{Z}}$, and the regions $\mathcal{R}$.

To compute the hybrid layout, we first compute a data or spatial layout $L_{\mathcal{Z}}$ on a coarse grid for the zones. The next step is then to compute the final, region-level layout $L_{\mathcal{R}}$.

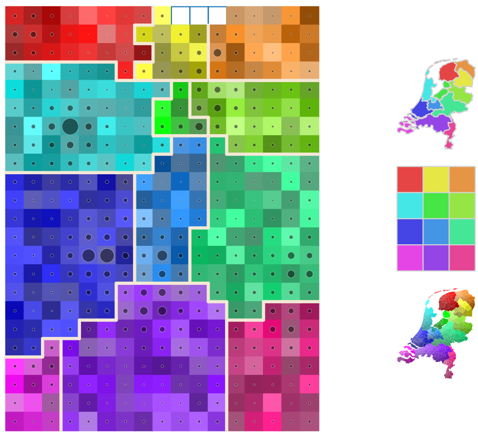
The difficulty lies with zones representing different numbers of regions. We want to obtain an outline for each zone in which to place and rearrange its regions. To this end, we leverage the algorithm for spatial layouts, by creating an artificial "map" based on the zone-level layout. Refer to Figure 10 for illustrations. We scale the regions of each zone $z_i$ such that it exactly fits the tile, translating it such that it is centered within the tile. Subsequently, we scale the regions further by a factor $\lambda < 1$. These steps give us the artificial map, for which we compute the spatial layout $L_s$ of all regions. If the region-level was assigned to be a spatial layout, we are done: $L_s$ is the final layout $L_{\mathcal{R}}$. If this level was assigned to be a data layout, then we compute a data layout for each zone $z_i$ separately, using the selected tiles for $z_i$ in $L_s$. The combination of these data layouts per zone then yields the final layout $L_{\mathcal{R}}$.

The second scaling step with $\lambda$ is done to ensure that each zone is represented by a compact, connected set of tiles. The parameter mostly needs to be sufficiently small, we use $\lambda = \frac{1}{W+H}$. For $\lambda$ to be closer to 1, the resulting zones could potentially preserve more of the shape of the original geography, at the expense of disconnected zones. However, due to the limited space available in the grid in computing the subsequent region-level layout, the gain is limited; when a data layout is used on the region level, the semantics of the preserved shape are lost and may even be misleading.



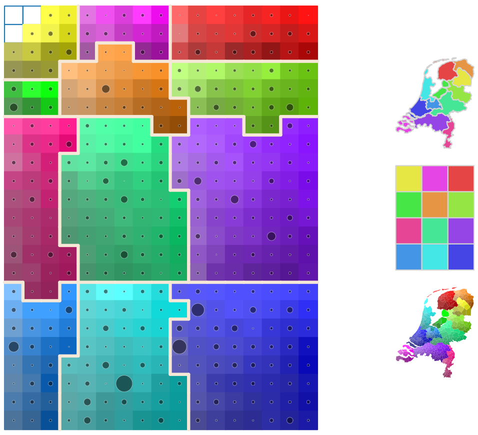
$S = 1.409$ $D = 0.004$

**(a)** Spatial $L_{\mathcal{Z}}$, spatial $L_{\mathcal{R}}$.



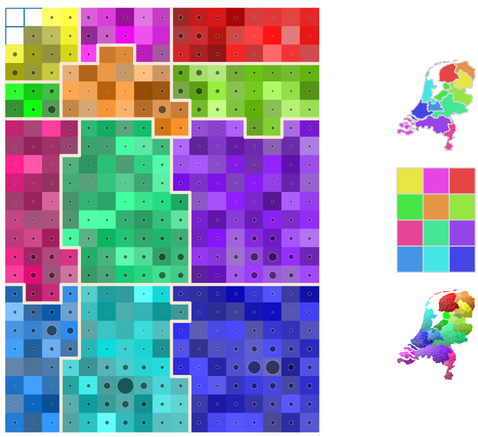
$S = 9.484$ $D = 0.427$

**(b)** Spatial $L_{\mathcal{Z}}$, data $L_{\mathcal{R}}$.



$S = 13.971$ $D = -0.022$

**(c)** Data $L_{\mathcal{Z}}$, spatial $L_{\mathcal{R}}$.



$S = 17.986$ $D = 0.408$

**(d)** Data $L_{\mathcal{Z}}$, data $L_{\mathcal{R}}$.

**Figure 12** Hierarchical grid maps for NL. With each, the smaller maps represent from top to bottom: the zones $\mathcal{Z}$, the zone layout $L_{\mathcal{Z}}$, and the regions $\mathcal{R}$.

**Results.** We demonstrate our techniques on our two hierarchical datasets in Figures 11 and 12. We assign each zone a hue, using a saturation and brightness gradient for that hue to color the regions within the zone. We render the boundary between tiles of different zones with a thicker line, to make it easier to identify zones. In both cases we see that our measures indicate that the spatial-spatial layout (a) has the lowest spatial distortion $S$, while having low data correlation $D$. The data-data layout (d) achieves the exact opposite.

Interestingly, for the EW dataset, the spatial-data layout (b) has worse spatial correlation than the data-spatial layout (c). Our measures, based on adjacent tiles, are sensitive to local structures being removed in reshuffling each zone, and measure only some form of spatial distortion for the relatively few tiles along the zone boundaries. The better higher-level spatial structure of (b) is not captured as much by our spatial distortion measure $S$, though it does seem to capture this structure for the NL dataset. Furthermore, it may have quite an influence on what an analyst may understand from the map. That is, it may be easier to still identify and relate the zones in a spatial-data layout, compared to doing so for regions within a zone, that are necessarily harder to identify due to their sheer number.

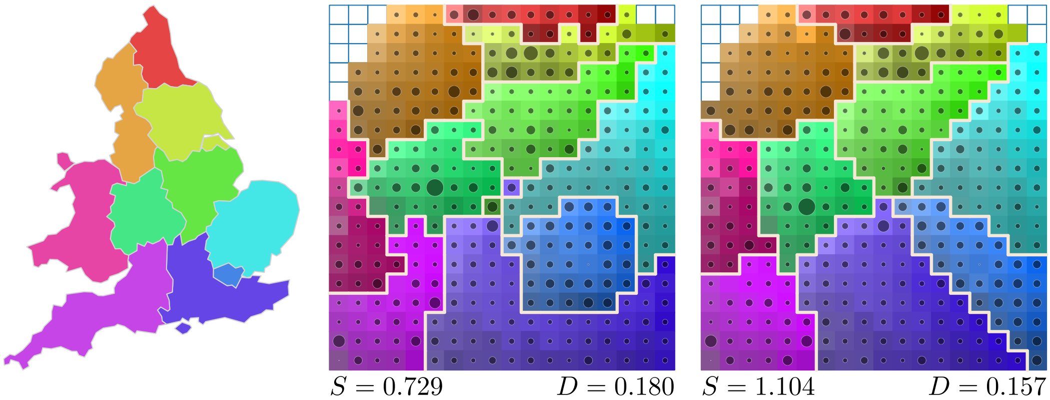



**Figure 13** Center: the spatial layout for the EW data, based only on the regions $\mathcal{R}$. Note the irregular zones and two disconnected zones. Right: a hierarchical layout computed by scaling regions towards their zone's geographical center.

For region-level data layouts $L_{\mathcal{R}}$ (b) and (d) we can observe uneven coloring in zones of rather similar data values, indicating unnecessary spatial distortion. Hence, instead of using a pure data layout, we could use a hybrid data-to-spatial layout with little slack. As we established in the previous section, such a layout will essentially retain the same data correlation and improve the spatial correlation significantly.

**Alternative for spatial zone-level layouts.** When using a spatial layout on the zone level, we aim to preserve the geographic relations of the zones. As such, the transformed map we compute in fact aims to resemble to original geography. We could indeed use the original geography – effectively using the non-hierarchical spatial layout for the entire map. Yet, this does have a different effect, as zones are now not represented as compactly. As an example, compare Figure 11(a) to Figure 13 (center): we observe the irregularly shaped zones in the latter, and even two zones that are not represented contiguously.

To promote compactness of zones, we could also opt to shrink the regions of a zone towards the zone's original, geographic centroid. Such an approach avoids the need to compute a zone-level layout. It naturally produces outlines roughly forming a Voronoi diagram of the centroids of the zones; see Figure 13 (right). However, it breaks the grid-like layout in our proposed solution which may help identifiability.

## 6 Discussion

We presented two approaches which allow us to combine spatial and data aspects in a grid map. Our hybrid grid maps use simulated annealing to effectively improve spatial or data correlation for a layout that is primarily based on the respective other aspect. Our experiments show that already a low amount of slack improves the other measure significantly, while not lowering the quality of the layout much with respect to the original measure – effectively also indicating that Tobler's law is indeed applicable here. Layouts with a large slack are usually inferior to layouts with a small slack computed from the other extreme. We believe that hybrid grid maps with low slack are not only visually pleasing, but also highlight patterns clearly and as such demonstrate the potential of our technique. It is, however, challenging to determine the cause of the complex patterns that arise in the grid maps: the patterns might be complex due to the complex nature of the problem, or due to imperfect grid allocation. Investigating this may be an interesting avenue for future research.

Furthermore, we introduced a controlled way of combining spatial and data aspects, by leveraging a hierarchy, assigning each level of the hierarchy to use a data layout or a spatial layout. This more enforced structure of mixing spatial and data aspects may aid in interpreting the resulting hybrid grid maps, as the zones act as logical units that stay together for higher-level patterns. Although we heuristically achieve a notion of connectivity by choosing a sufficiently small $\lambda$, we do not enforce this constraint. In future work, extra constraints may be added in the spatial layout algorithm to ensure connectivity, along the lines of the work by Validi et al. [23]. Though hierarchical grid maps readily generalize to hierarchies of multiple levels, we expect that repeatedly changing from data to spatial layouts and vice versa may result in layouts that are too complex to be understood.

**Measures.** The quality measures that we used to assess spatial and data quality encapsulate our main goal of creating smooth color and data changes along the grid.

It would be interesting for future research to investigate how well these notions of spatial and data correlation match the expectations of a user, or how effective different measures are for predicting task performance on hybrid grid maps and hierarchical hybrid grid maps.

**Usability.** The main question looking forward is whether such hybrid layouts indeed help synthesizing a mental model of the data. That is, can a human analyst effectively work with hybrid layouts? With these methods we have shown that it is now possible to create such hybrid layouts, and hence these questions may now be further researched.

Intuitively, it seems that spatial layouts with some improved data correlation are useful. It erases some of the “noise” in the data dimension at the cost of slight distortion of space – but as grid maps are inherently distorted, such seems tolerable and inherent in the approach to begin with. The other end, data layouts with some improved spatial correlation, can also be useful and it is feasible to significantly reduce the spatial distortion while still maintaining a layout with high data correlation. Hierarchical hybrid layouts are more constrained, but for that reason also offer more control in creating such layouts with a strong mix of both aspects. They may hence be easier to interpret than basic hybrid layouts with medium levels of slack.

We color the tiles by their spatial location, to communicate spatial correlation. However, we observe that there may be visual bias for regions with large data values or bright colors. When such regions are contrasting their adjacent tiles, this is more apparent and feels more out of place than when less bright or smaller data values are out of place. The visual assessment is further deceived by an element of the coloring scheme we use for non-hierarchical cases. We rotate the hue around a point in the map. Near this point, regions might be close together while having a different hue, and hence seem far apart in the coloring. We aimed to reduce this effect be reducing brightness and saturation near the center. As an alternative to color, interaction could help in identifying regions.

An eventual hybrid grid map may hence require further attention as to how spatial distortion is communicated and how data values are rendered. A prominent question is to what level of detail spatial distortion should be indicated. Standard grid maps often do not communicate their distortion at all. Yet, in a hybrid layout, being able to separate spatial from data patterns and effects may increase the need for indicators of distortions.

## References


1. Bonnie Berkowitz and Lazaro Gamio. What you need to know about the measles outbreak, February 2015. Accessed January 2023. URL: `https://www.washingtonpost.com/graphics/health/how-fast-does-measles-spread/`.
2. Paul Blickle and Sascha Venohr. Dürfen wir vorstellen: Deutschlands Muslime, January 2015. Accessed January 2023. URL: `http://www.zeit.de/gesellschaft/2015-01/islam-muslime-in-deutschland`.

3 Mark Bruls, Kees Huizing, and Jarke J. van Wijk. Squarified treemaps. In *Proceedings of the Joint EUROGRAPHICS and IEEE TCVG Symposium on Visualization*, pages 33–42, 2000. doi:10.1007/978-3-7091-6783-0_4.

4 Kevin Buchin, Bettina Speckmann, and Sander Verdonschot. Evolution strategies for optimizing rectangular cartograms. In *Proceedings of the 7th International Conference on Geographic Information Science*, LNCS 7478, pages 29–42, 2012. doi:10.1007/978-3-642-33024-7_3.

5 Rafael G. Cano, Kevin Buchin, Thom Castermans, Astrid Pieterse, Willem Sonke, and Bettina Speckmann. Mosaic drawings and cartograms. *Computer Graphics Forum*, 34(3):361–370, 2015. doi:10.1111/cgf.12648.

6 Ben Casselman and Allison McCann. Where your state gets its money, April 2015. Accessed January 2023. URL: http://fivethirtyeight.com/features/where-your-state-gets-its-money/.

7 Mark de Berg, Elena Mumford, and Bettina Speckmann. Optimal BSPs and rectilinear cartograms. *International Journal of Computational Geometry & Applications*, 20(2):203–222, 2010. doi:10.1142/S0218195910003268.

8 Danny DeBelius. Let's tesselate: Hexagons for tile grid maps, May 2015. Accessed January 2023. URL: http://blog.apps.npr.org/2015/05/11/hex-tile-maps.html.

9 David Eppstein, Marc van Kreveld, Bettina Speckmann, and Frank Staals. Improved grid map layout by point set matching. *International Journal of Computational Geometry & Applications*, 25(02):101–122, 2015. doi:10.1142/S0218195915500077.

10 Diansheng Guo, Jin Chen, Alan M. MacEachren, and Ke Liao. A visualization system for space-time and multivariate patterns (vis-stamp). *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1461–1474, 2006. doi:10.1109/TVCG.2006.84.

11 Wouter Meulemans, Jason Dykes, Aidan Slingsby, Cagatay Turkay, and Jo Wood. Small multiples with gaps. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):381–390, 2016. doi:10.1109/TVCG.2016.2598542.

12 Wouter Meulemans, Max Sondag, and Bettina Speckmann. A simple pipeline for coherent grid maps. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1236–1246, 2020. doi:10.1109/TVCG.2020.3028953.

13 Patrick A.P. Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37(1/2):17–23, 1950. doi:10.2307/2332142.

14 New York Times. How the rulings affect gay couples, June 2013. Accessed January 2023. URL: http://www.nytimes.com/interactive/2013/06/26/us/scotus-gay-marriage.html.

15 Kenton Powell, Rich Harris, and Feilding Cage. How voter-friendly is your state?, October 2014. Accessed January 2023. URL: http://www.theguardian.com/us-news/ng-interactive/2014/oct/22/-sp-voting-rights-identification-how-friendly-is-your-state.

16 Ben Shneiderman and Martin Wattenberg. Ordered treemap layouts. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 73–78, 2001. doi:10.1109/INFVIS.2001.963283.

17 Aidan Slingsby. Tilemaps for summarising multivariate geographical variation. In *Proceedings of the Workshop on Visual Summarization and Report Generation*, 2018.

18 Aidan Slingsby, Mary Kelly, and Jason Dykes. OD maps for showing changes in Irish female migration between 1851 and 1911. *Environment and Planning A*, 46(12):2795–2797, 2014. doi:10.1068/a140112g.

19 Aidan Slingsby, Jo Wood, and Jason Dykes. Treemap cartography for showing spatial and temporal traffic patterns. *Journal of Maps*, 6(1):135–146, 2010. doi:10.4113/jom.2010.1071.

20 Waldo R. Tobler. A computer movie simulating urban growth in the Detroit region. *Economic geography*, 46(sup1):234–240, 1970. doi:10.1111/j.1467-8306.2004.09402005.x.

21 Alex Tribou and Keith Collins. This is how fast America changes its mind, June 2015. Accessed January 2023. URL: http://www.bloomberg.com/graphics/2015-pace-of-social-change/.

22 Edward R. Tufte. *The Visual Display of Quantitative Information*, volume 2. Graphics Press Cheshire, CT, 1983. doi:10.1075/idj.4.3.12cos.

23 Hamidreza Validi, Austin Buchanan, and Eugene Lykhovyd. Imposing contiguity constraints in political districting models. *Operations Research*, 70(2):867–892, 2022. doi:10.1287/opre.2021.2141.

24 Marc J. van Kreveld and Bettina Speckmann. On rectangular cartograms. *Computational Geometry: Theory and Applications*, 37(3):175–187, 2007. doi:10.1016/j.comgeo.2006.06.002.

25 Jo Wood, Donia Badawood, Jason Dykes, and Aidan Slingsby. BallotMaps: Detecting name bias in alphabetically ordered ballot papers. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2384–2391, 2011. doi:10.1109/TVCG.2011.174.

26 Jo Wood and Jason Dykes. Spatially ordered treemaps. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1348–1355, 2008. doi:10.1109/TVCG.2008.165.

27 Jo Wood, Jason Dykes, and Aidan Slingsby. Visualisation of origins, destinations and flows with OD maps. *The Cartographic Journal*, 47(2):117–129, 2010. doi:10.1179/000870410X12658023467367.

28 Jo Wood, Aidan Slingsby, and Jason Dykes. Visualizing the dynamics of London's bicycle-hire scheme. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 46(4):239–251, 2011. doi:10.3138/carto.46.4.239.