Author's copy. To appear in Proceedings of 14th IEEE Pacific Visualization Symposium (PacificVis 2021).

On the Visualization of Hierarchical Multivariate Data

Boyan Zheng*
Heidelberg University, Germany

Filip Sadlo†
Heidelberg University, Germany

Abstract

In this paper, we study the visual design of hierarchical multivariate data analysis. We focus on the extension of four hierarchical univariate concepts—the sunburst chart, the icicle plot, the circular treemap, and the bubble treemap—to the multivariate domain. Our study identifies several advantageous design variants, which we discuss with respect to previous approaches, and whose utility we evaluate with a user study and demonstrate for different analysis purposes and different types of data.

Index Terms: Human-centered computing—Visualization—Visualization techniques—Treemaps; Human-centered computing—Visualization—Visualization application domains—Information visualization

1 Introduction

Multivariate data and hierarchical data are ubiquitous. But whereas visualization has researched either field intensely and very successfully during the last decades, only very little is known about the visualization of hierarchical multivariate data, i.e., trees with multiple attributes at each node. However, since such data are broadly present, for example, as omics data in biology, statistics data in census, or business data of hierarchical organizations, dedicated visualization techniques are needed for their effective analysis.

This motivates us to carry out a systematic design study on the visualization of multivariate quantitative features with a common hierarchical structure. Based on task analysis, we narrow down the design space to some promising combinations of techniques from multivariate visualization and hierarchical visualization. Guided by the requirements and respective motivations, we finally derive four effective visual representations for hierarchical multivariate data: multivariate sunburst charts (MSB), multivariate icicle plots (MIP), multivariate circular treemaps (MCT), and multivariate bubble treemaps (MBT), which we evaluate using a user study and discuss in terms of their strengths and limitations.

The contributions of this paper include:

- systematic study on visualizing hierarchical multivariate data,
- four new visual solutions (MSB, MIP, MCT, MBT),
- evaluation of the proposed solutions by a user study, and
- optimization of the bubble treemap layout algorithm.

2 Related Work

There exist different classification schemes for visualizing multivariate data. Keim and Kriegel [18, 19] classify them into six types: pixel-oriented, geometric, icon-based, hierarchical, graph-based, and hybrid. Notice that the hierarchical type is considered for multiple ordinal attributes, i.e., hierarchical structure, whereas we focus on quantitative multivariate data. Wong and Bergeron [42] divide the techniques into two-variate display-based, multivariate display-based, and animation-based. He et al. [16] provide a problem-oriented classification, which groups the techniques by the tasks "feature", "fusion", and "correlation". There exist also numerous hierarchical visualization techniques, a collection of which is presented by Schulz [30]. Basically all these tree-based visualization techniques are derived from the following five basic visualization approaches: node–link diagrams, treemaps, circular treemaps, icicle plots, and sunburst charts. To the best of our knowledge, there are very few works that correlate these two fields.

Several previous works have proposed visualization approaches for hierarchical multivariate data for very specific applications [8, 10, 12, 21, 28], which are, however, not effective for static visualization or general cases. Wittenburg et al. [40, 41] present approaches for visualizing hierarchical multivariate data by aligning the cells in treemaps. In our work, we do not include treemaps, because of their issues with reading the hierarchy and the constraints on showing internal nodes [25, 43]. Another example for hierarchical multivariate visualization are the hierarchical parallel coordinates by Fua et al. [13], which, however, serve for hierarchical exploration of multivariate data, but cannot convey the hierarchical structure of the data. Engel et al. [11], on the other hand, present an interesting approach to transform multivariate data into a tree. Nevertheless, our goal is not to "bridge" multivariate and hierarchical representations, but to integrate them. Vosough et al. visualize hierarchical categories in parallel coordinates [36], which addresses visualization of multiple hierarchies of multivariate data. In their case, each attribute has a different hierarchical category. In contrast, we focus on multivariate data with a common hierarchy shared by all attributes. Arleo et al. present GiViP [2], which can reveal the inter-relationships between hierarchical attributes based on a redesigned chord diagram. However, our focus is on quantitative analysis of multivariate nodes within a common hierarchy.

3 Task Analysis

Before we describe our design choices (Section 4) and the visual design (Section 5), let us motivate our work.

3.1 Hierarchical Multivariate Data

The terms *multivariate* and *multidimensional* are often vaguely used. Nevertheless, in our context, the term multivariate refers to the dimensionality of the range, while the term multidimensional refers to the dimensionality of the domain [4]. As an extension of univariate and bivariate, the term multivariate (or hypervariate) typically refers to data that consist of three or more variables or observations. In this study, we focus only on data that are multivariate in terms of quantitative level of measurement, and are discrete. Second, the data of interest are structured under a defined hierarchy. Thus, hierarchical multivariate data are data that exhibit a tree structure, with all nodes (including internal ones) consisting of multiple attributes, or in other words, multiple attributes share the same tree structure. Mathematically, they are very similar to a directed tree

$$T(V,E), \tag{1}$$

(with nodes V and edges E), with the only difference that the nodes possess multiple attributes. Notice that we therefore use the term "hierarchical multivariate" for this kind of data—as opposed to the term "multivariate hierarchical" data.

*e-mail: boyan.zheng@iwr.uni-heidelberg.de
†e-mail: sadlo@uni-heidelberg.de

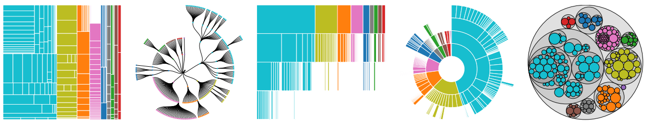

Figure 1. Traditional visualization concepts for hierarchical data. Treemaps (a), radial trees (b), icicle plots (c), sunburst charts (d), and circular treemaps (e), visualizing the Flare dataset using D3 [7].

3.2 Open Challenges

Generally, tree/hierarchy visualizations can be grouped into two main types of design—approaches based on explicit node–link representations, and approaches employing space-filling strategies which are implicit [31]. In traditional node–link representations, visualization of even a single attribute per node is nontrivial, as illustrated in Figure 1b. In traditional space-filling strategies, on the other hand, two attributes could be mapped to visual variables, e.g., to size/length and color, as demonstrated in Figure 1c. There exist many approaches to visualize additional attributes in traditional hierarchical representations. They typically employ additional patterns, as in the case of uncertainty [15, 32], or features such as height [39], which results in 3D representations with the involved difficulty of occlusion. However, such approaches are still very limited regarding the number of additional data range dimensions. Noted by Nobre et al. [27], tree visualizations support very poorly on representing several ($\geqslant 5$) node attributes. The main goal of this work is to tackle this challenge in 2D visual space. From our results (Section 7), we can see that our techniques are able to visualize seven attributes. Another open challenge is with respect to representation of arbitrary data on internal nodes. Most tree visualization techniques follow an aggregation scheme, i.e., the internal nodes in the treemap, the icicle plot, and the sunburst chart show the sum of their children. The traditional circular treemap (Figure 1e), in contrast, just packs children into circles whose radii do not represent quantities. The node–link diagram, on the other hand, can show any data on internal nodes, but as mentioned before, even showing one attribute is challenging. The goal of this work is also aiming to break the constraints of hierarchical visualization techniques. Our resulting MSB, MIP, and MCT address this open challenge.

3.3 Requirements

Since this research is inspired by Functree2 [10] from the field of bioinformatics, we have conducted several face to face discussions with bioinformaticians who work on omics data. In these discussions, we presented them Functree2, and asked them for their assessment of the utility of Functree2 and possible improvement, with the aim of deriving general requirements on visualizing hierarchical multivariate data. They mentioned, for example, that the nodes in Functree2 are hard to see due to their small size, which inspired us to investigate possible alternatives with higher space-efficiency. Since they agreed that Functree2 is easy to understand, and due to the fact that domain experts are typically not visualization experts, we also want possible alternatives to be simple to interpret. In terms of visual encoding, the mapping suggestions proposed by Mackinlay [23] have been considered. As an initial work on this topic, and to serve for print publications, we limit the scope to 2D static representations. Furthermore, the domain experts mentioned that Functree2 could help them with comparison tasks on all levels, which became our main focus. Additionally, they also mentioned some specific requirements that might not be easy to generalize, such as depicting arbitrary data on internal nodes. We realized that in almost all tree visualization techniques, the internal nodes only show aggregation of the children's values, or even no data (Section 3.2). Thus, for hierarchical multivariate data in general, we rank this requirement with the least priority in terms of generalizability. Based on these rationales, we determined the list of general desirable properties for the resulting approaches:

- R1 *Static representation*: applicable for (printed) publication;
- R2 *Avoid occlusions*: limit the representation to 2D;
- R3 *Simple interpretation*: explicit visual mapping;
- R4 *Readability*: clear multivariate and hierarchical properties;
- R5 *Arbitrary data on internal nodes*: not just aggregation.

We put R5 last due to its least priority.

4 Design Choices

4.1 Visualization of Hierarchical Data

When visualizing hierarchical data, in limited visual space, there is always a trade-off between compactness and readability of the hierarchy [15]. Hierarchical visualizations with very low compactness, such as node–link diagrams (e.g., the radial tree in Figure 1b), may profit from increased readability of the hierarchy [24], but utilize too much visual space [22], limiting the amount of space available for visualizing multivariate properties, which decreases the readability of the multivariate properties, as shown in Figure 6a. On the other hand, implicit hierarchical visualizations are highly space-efficient. However, very compact visualizations, as for example the treemap (Figure 1a), achieve a high information density, but typically make it difficult to read the hierarchical structure of the data [25, 43], with the result that Requirement R4 from Section 3.3 is not fulfilled. In our choice of hierarchical visualization concepts, we need to balance the compactness and the readability of the hierarchy. Thus, we do neither favor the treemap nor the node–link diagram concept as candidates for conveying hierarchical structure.

Hierarchical visualization techniques that may exhibit occlusions, like Beamtrees [34] or Cheops [3], do not fulfill Requirement R2 to avoid occlusions. Out of the five basic and well-known tree visualization techniques shown in Figure 1, we consider the icicle plot (Figure 1c), the sunburst chart (Figure 1d), and the circular treemap (Figure 1e) as potential candidates for integration with multivariate visualization concepts, since they balance readability and compactness [22, 25]. In our considerations (and Table 1), we treat icicle plots and sunburst charts as being related, as the latter is a circular variant of the former. Although related, we take both into consideration, since icicle plots have advantages on accuracy and efficiency in reading, and sunburst charts have advantages regarding user acceptance [33]. We denote the smallest unit (small rectangle) in the icicle plot and ("curved rectangle") in the sunburst chart as a *segment*. Notice that in the case of the icicle plot, the horizontal extent of a segment is denoted *width* in this paper, and in the sunburst chart, the angular extent of a segment is denoted *angular width*. That is, e.g., the angular widths of the direct children of the root node in a sunburst chart sum up to 2π. In both the sunburst chart and the icicle plot, we denote the other extent of a segment to be its *length*, which is constant in both schemes (radial length in the sunburst chart and vertical length in the icicle plot).

Furthermore, in order to fulfill Requirement R5, proper adjustments are clearly needed. Notice that, since we base our techniques on traditional tree visualization techniques, our techniques inherit their limitation on scalability, e.g., regarding the representation of large and deep hierarchical structures.

4.2 Visualization of Multivariate Data

For multivariate visualization, candidate concepts need to utilize the visual space left by the trade-off between compactness and readability of the hierarchy. For example, parallel coordinates, scatterplot matrices, and dimensional stacking [20] would be hard to combine with hierarchical approaches because they already exhibit a quite high visual complexity and require a considerable amount of space. Glyphs, as a class of multivariate visualization concepts that uses small and independent visual objects [6], have their major strength in readability [38], making them our targeted candidate.

Many glyph concepts do not lend themselves well for multivariate visualization under the requirements we have generalized in Section 3.3 [14]. For example, techniques like Chernoff faces [9] and stick figures [29] do generally not map to strong quantitative visual variables [23], and are therefore not satisfying Requirement R3. Using bar charts for the multivariate attributes, on the other hand, would further decompose the width of segments, which already tends to be too thin in hierarchical visualization, as discussed below. This motivated us to use stacked bar charts instead. Clock glyphs could be a candidate, but since area is a better quantitative visual variable than color, we chose pie charts instead. Star glyphs / radar charts could be an option as well, but their line-type design would make it difficult to identify attributes.

As a result of this search, we identify the stacked bar chart and the pie chart as those multivariate visualization concepts, that are sufficiently compact and simple to read, and thus good candidates for integration with hierarchical visualization techniques. Notice that such concepts also suffer from visualizing large numbers of attributes, which limits their scalability.

4.3 Design Space

Based on the previous discussions, Table 1 lists the design space within which this study is located, with the orange-marked area being determined as potentially promising. Since the potential glyph candidates need to follow the layout placement driven by the hierarchical component [38], the aspect of their outer shape must be considered as well. That is, in order not to lose compactness, we do not want to combine rectangular concepts with circular ones, e.g., to combine (stacked) bar charts with circular treemaps, or pie charts with sunburst charts / icicle plots. Thus, the two remaining combinations we examine are stacked bar charts included in sunburst charts and icicle plots, and pie charts included in circular treemaps.

Next, we document our visual design, guided by the previously determined requirements, and located in the considered design space area from Table 1. Our design is structured into two main parts: multivariate sunburst charts and multivariate icicle plots (Section 5.1), and multivariate circular treemaps and multivariate bubble treemaps (Section 5.2).

5 Visual Design

5.1 Multivariate Sunburst Chart / Multivariate Icicle Plot

As a basis for our integration, let us now investigate the properties and limitations of the traditional sunburst chart and traditional icicle plot in more detail (cf. Figure 1d and 1c):

P_a1 **Inconsistent mapping in sunburst charts.** The sunburst chart maps the quantities (the data at the leaves of the hierarchy and the sum of the children at internal nodes) to the angular width of each segment. That is, the (radial) length of all segments is constant, i.e., all segments are aligned on circles, and these circles are uniformly spaced (white circles in Figure 1d). Since the perimeter of these circles varies with radius, the sunburst chart does not map consistently to the visual variable "area". (For icicle plots, value maps to the width of a segment and thus consistently to the visual variable "area".)

P_a2 **Medium compactness.** If a node has no children, then the angular width (sunburst chart) or width (icicle plot) of the parent is not used in descendant layers, causing unused visual space.

P_a3 **Thin leaves.** Because descendants can only inherit from the angular width of the parent, "deep" descendants can become very thin in the sunburst chart (e.g., pink in Figure 1d). If we additionally delineate segments with (white) outlines of constant width, sufficiently deep nodes cannot be visualized at all. The icicle plot suffers even more from this issue, as the segment width, in contrast to the sunburst chart, does not grow with deeper levels (Figure 1c).

P_a4 **Aggregation.** Both sunburst charts and icicle plots can only show the sum of the children's values at internal nodes.

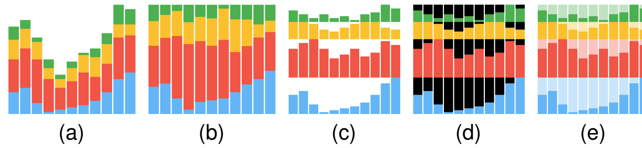

Figure 2. Design study for the stacked bar chart component. Traditional stacked bar chart (a), and its normalized alternative (b) for eased attribute comparison. Separated alignment (c), padded with background color, could lead to visual ambiguities when combined with hierarchical techniques. Padding with another color (d) leads to visual clutter and can cause misinterpretation. Padding with low-saturated color (e) is our favored design component.

The design space led us to incorporate stacked bar charts into the sunburst chart / icicle plot, with each segment of the sunburst chart / icicle plot containing one stacked bar. The first design task is the orientation of the bars. To prevent asymmetry in the representation and to avoid mapping of quantities to angular width, which is not a good quantitative visual variable, we orient the stacked bar chart in radial (sunburst chart) or vertical (icicle plot) direction, i.e., value is encoded in direction of the (radial) length of the segment.

Straightforward application of traditional stacked bar charts (Figure 2a) would make it, however, hard to compare an attribute between siblings (between individual bars of the stacked bar chart). This motivated us to align the stacked portions, which we first tried to achieve by normalizing each bar (Figure 2b). Although such normalization eases comparing an attribute across different bars, it complicates quantitative interpretation. Therefore, our second (and preferred) approach is to align each attribute separately (Figure 2c–2e). Our first design (Figure 2c) works quite well with an individual stacked bar chart. However, it would not work well if integrated into the sunburst chart / icicle plot, because there, we also need (white) area for separating the nodes, as well as the levels of the hierarchy (corresponding to the white circles from Figure 1d), which would lead to visual interference. Thus, we experimented with another "padding" color in the stacked bar chart, for example, black (Figure 2d), but this visually disrupted the stacked information, making it harder to perceive it as a multivariate entity. Beyond that, the padding color could be misunderstood as an additional attribute. All this motivates us to use low-saturated color for padding (Figure 2e), which we choose as the solution for our integrated approach.

As already motivated, we put the stacked bars in vertical direction into the icicle plot (Figure 3a), and in radial direction into the sunburst chart (Figure 3c). This means that, in contrast to the tradi-

Table 1. Considered design space in this study (orange), chosen solutions (checked), and related work. The design space is spanned by multivariate visualization (MV) techniques, and hierarchical visualization (HV) techniques. CT stands for circular treemaps, SB for sunburst charts, IP for icicle plots, and NLD for node–link diagrams.

HV / MV	CT	SB/IP	treemaps	NLD
bar charts	[8]		[40,41]	
stacked bar charts		✓		[10,21]
pie charts	✓			[21]
star glyphs				
clock glyphs	[12]			

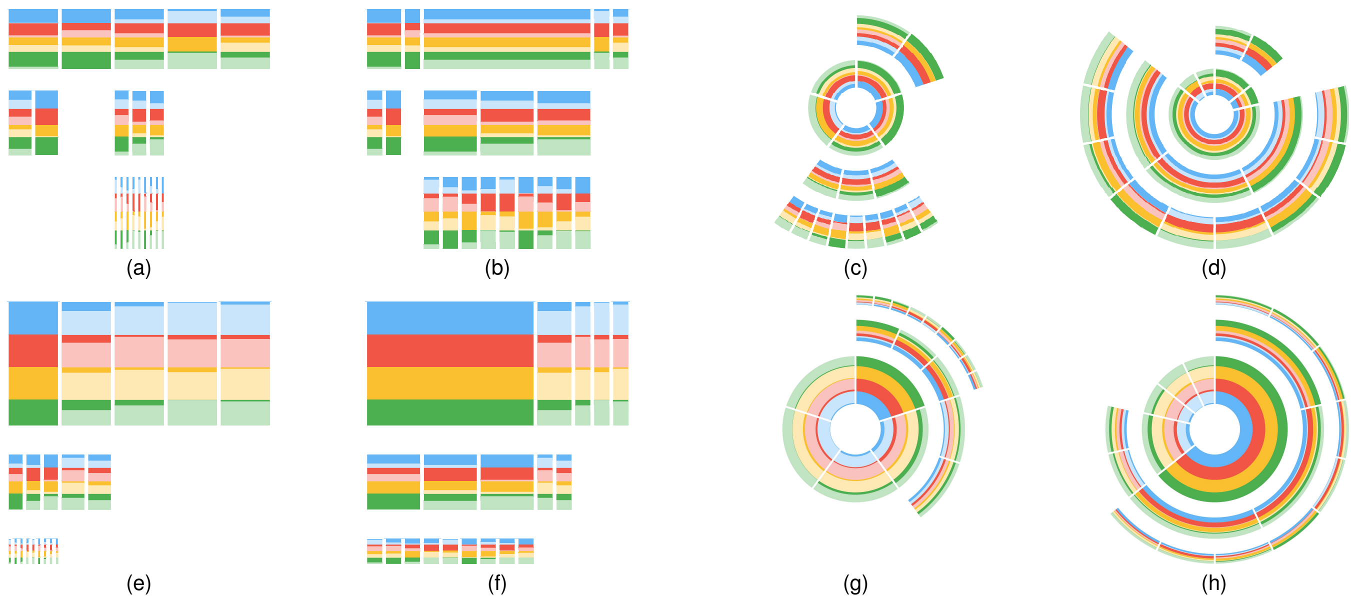

Figure 3. Multivariate icicle plot (left two columns) and multivariate sunburst chart (right two columns), demonstrated using the same simple synthetic dataset. Internal nodes showing the average of their children (top row), and showing the sum of their children (bottom row). Layout strategy with equal sibling (angular) width ((a), (e), (c), (g)), and with equal leaf (angular) width ((b), (f), (d), (h)).

tional sunburst chart, the quantitative information is mapped to the radial length of a segment, instead of mapping to the angular width, and in case of the icicle plot, it is mapped to the length of a segment instead of its width. This is a necessary design decision, since both traditional sunburst charts and traditional icicle plots can only show the sum of the children's values at internal nodes (P_a4) using (angular) width. However, by this new mapping in our resulting concept, any multivariate data can be shown at internal nodes.

Thus, we are now free to choose the (angular) width of the segments. We determined two useful strategies for choosing the (angular) width: the *equal siblings strategy* and the *equal leaves strategy*. The equal siblings strategy takes the (angular) width of the parent node and distributes it equally to the (angular) widths of its children. See the strategy applied to the sunburst chart in Figure 3c and 3g, and to the icicle plot in Figure 3a and 3e. The equal leaves strategy, in contrast, distributes the entire global (angular) width (2π in case of the sunburst chart, and the entire visualization width in case of the icicle plot) equally to all leaf segments of the hierarchy. See Figure 3d and 3h for this strategy applied to the sunburst chart, and Figure 3b and 3f for the icicle plot. It is apparent that the equal leaves strategy successfully avoids the "thin leaves" issue (P_a3), increases the compactness (P_a2), and eases the identification and reading of leaves [24]. However, one can also see that with the equal siblings strategy, the hierarchy and internal nodes are better readable. With both strategies, both the hierarchy aspect and the multivariate aspect of the hierarchical multivariate data are well conveyed. Finally, Figure 3 also demonstrates that our approaches are able to show on an internal node the sum of the values of its children, but also any other quantity, as demonstrated with the mean value of the children. This concludes the design of our MSB and MIP.

5.2 Multivariate Circular Treemap / Multivariate Bubble Treemap

The second part of our design study focuses on extending the concept family of circular treemaps to the multivariate domain. Let us summarize the properties and limitations of the traditional circular treemap (Figure 1e) first:

P_b1 **Consistent mapping.** The traditional circular treemap maps the quantities at the leaves to the area of disks. Thus, the circular treemap maps consistently to the visual variable "area".

P_b2 **Medium compactness.** Since the parent nodes are circles too, the compactness is not very high, but it is also not as low as, e.g., in the case of the radial tree.

P_b3 **Depth encoding by saturation.** To improve the readability, the depth of a node is typically mapped to brightness, hue, or saturation of the background color of the respective circle, as demonstrated in Figure 1e. However, due to perceptual limitations, this constrains the available maximum number of hierarchy levels.

P_b4 **No information on internal nodes.** The circular treemap does not show any quantitative data on internal nodes. They only present the hierarchical information by grouping the representations of their children. Thus, only the leaves show quantitative information.

The design space guided us to choose the pie chart to be incorporated into the circular treemap. In case of the sunburst chart / icicle plot, our first design decision was about the orientation of the stacked bar within each segment. In case of the circular treemap, pie charts fit perfectly into a disk-based visual representation scheme, and due to symmetry, orientation is not an issue.

The next design step again investigates the multivariate visual component itself, which is the pie chart in our case. Straightforward application of the traditional pie chart (Figure 5a) exhibits some limitations. In the traditional pie chart, each quantitative attribute is mapped to a sector of the disk, and the area of the disk depends on the sum of the values of all attributes. If the attributes have different units, however, such a representation would not be suitable. Nevertheless, it is also not readily visible that, e.g., the red and light blue sectors in Figure 5a exhibit the same value, and how much the yellow and purple sectors differ.

This motivated us to search for alternatives and optimizations based on pie charts. We found polar area diagrams (also known as nightingale rose charts). Instead of mapping the quantities to sectors with varying angle, polar area diagrams map each quantity to a sector with equal angle but varying radius, as shown in Figure 5b. We extended the polar area diagram with a circular guide to sup-

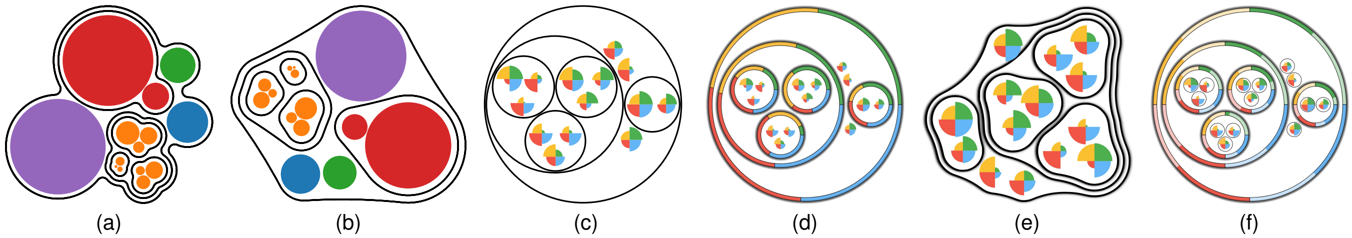

Figure 4. Hierarchy of the dataset from Figure 3, visualized with a bubble treemap with internal nodes by concave hulls (a), and internal nodes by almost convex hulls (b). Entire synthetic dataset from Figure 3: (c) example state after step 1 of the bubble treemap algorithm, (d) MCT with each internal node showing the multivariate mean of its children, (e) MBT, and (f) MCT for data with different units.

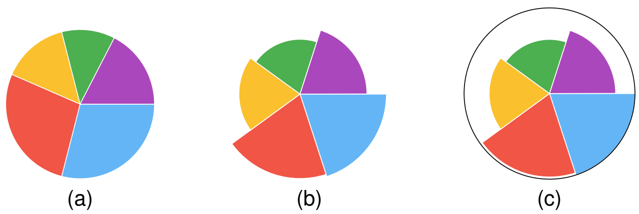

Figure 5. (a) In pie charts, the area of the disk is determined by the sum of all attributes, and the area of a sector by its value. (b) In polar area diagrams, the angle is fixed, and the value is mapped to area. This enables visualization of attributes with different units, in contrast to (a). (c) Added maximum guide for eased comparison.

port quantitative comparison with its maximum (Figure 5c), which is applicable in cases where all attributes have the same unit or if attributes with different units have been normalized. While both the pie chart and the polar area diagram map quantity to area, polar area diagrams have several benefits:

- **Simple localization of attributes.** Since the angles in polar area diagrams are uniform and thus fixed, it is much easier to locate a certain attribute, compared to traditional pie charts, where sectors have varying angles and varying offsets.
- **Simple comparison.** Polar area diagrams ease intra-node and inter-node comparison. The visual variable "length", which is related to polar areas, helps to compare different attributes within a node. Moreover, the additional guide from Figure 5c helps to judge a value in its range. Inter-node comparison is eased, due to the simplified localization of attributes.
- **Simple identification of small values on leaves.** The equal angle strategy in polar area diagrams makes each sector well visible. Small values on leaves lead to short sectors instead of very thin ones (as in sunburst charts / icicle plots), which are easier to spot.

Based on this evaluation, we identified polar area diagrams (with guide) as the best candidate for integration into circular treemaps.

As a design alternative to increase the compactness of the hierarchical scheme, we also take into consideration the bubble treemap [15] (Figure 4a and 4b), which uses compact packing curves for reflecting the hierarchy, instead of circles, as used in the circular treemap. In the bubble treemap, the smoothness of an internal node's packing curve can be adjusted by a parameter, providing shapes ranging from convex hulls to concave hulls. The increased compactness of the bubble treemap enables larger leaf representation, as exemplified in the comparison between our resulting approaches, the MCT (Figure 4d) and MBT (Figure 4e).

However, both the MCT and the MBT so far share the limitation that they cannot show quantitative data on internal nodes (P_b4). Thus, it is the final goal of our design study to provide a solution to this limitation, and we found such a solution for the MCT. We redesigned the circular packing ring, which groups the children of internal nodes, and visualize the data of such nodes on that ring. Figure 4d shows an example, where we visualize on internal nodes the multivariate average of their children. In case of different units, we split the ring into equal-angled sectors, one sector for each attribute, and display the attribute (normalized with respect to the range of the attribute) with saturated color in that sector, padded with low-saturated color, similar to our stacked bars, as shown in Figure 4f. However, a side effect of this approach is that the thick circles (leveraged internal nodes) affect the compactness and squeeze the visual space of the leaves.

Finally, the traditional circular treemap uses increasing, e.g., saturation of the internal node disks to represent the depth and thus improve readability (P_b3). However, due to perceptual limitations, this limits the maximum depth of the hierarchy. Additionally, the colors of the internal node disks would likely lead to interference with the polar area diagrams. We researched this problem and came up with the option to add narrow shadows on the outer boundary of the disks, which is similar to the idea in Cushion treemaps [35]. We employ this approach in both of our solutions, the MCT and the MBT (Figure 4d and 4e) to support readability.

6 Implementation

As the implementation of our MSB, MIP, and MCT is rather simple and straightforward, we only provide some details on the implementation of the MBT.

Our main contribution regarding the implementation of the multivariate bubble treemap is with respect to optimization of the original algorithm to compute the bubble treemap [15]. Compared to the original algorithm in [15], we employ one additional step of sorting for acceleration of the computation of the force-based layout. The overall process structures as follows (Algorithm 1):

1. Determine the largest radius among all attributes for each node, and compute the traditional circular treemap layout. This can be achieved using various algorithms [37,44]. In our implementation, we use the circle packing library in D3 [7].
2. Sort all nodes in the tree by depth in descending order.
3. Apply a fast force-based layout algorithm.
4. Replace all disks with corresponding polar area diagrams.

To compute the layout, our algorithm runs the force simulation once for all nodes at the same depth of the tree, whereas the original bubble treemap algorithm needs to run the force simulation once for every internal node. Thus, the time complexity of our algorithm is $O(d(T))$, with $d(T)$ being the depth of the tree, in contrast to the original bubble treemap algorithm's time complexity $O(i(T))$, with $i(T)$ being the number of internal nodes in the tree.

7 Case Study

To demonstrate the properties of our approaches and their utility, we start our evaluation with a case study using omics data (Section 7.1), whose attributes exhibit the same unit. Our second application is on statistics data (Section 7.2), whose attributes have different units.

Algorithm 1 MBT layout

```
Require: nodes = {n}, ∀n ⟹ n.depth ↓
procedure FASTLAYOUT(nodes)
    for all n ∈ nodes do
        if n.depth has changed then
            RUNSIMULATION(S)
        end if
        if n.height is 0 then
            circle ← CIRCLE(n.center, n.radius)
            circle.pid ← n.parent.id
            string ← STRING(circle, n.parent.center)
            string.pid ← n.parent.id
            set proper force on string
            add circle, string to scene S
        else
            build hull for all circles and hulls whose pid is n.id
            hull.pid ← n.parent.id
            delete all string in S whose pid is n.id
            string ← STRING(hull, n.parent.center)
            string.pid ← n.parent.id
            set proper force on string
            add hull, string to scene S
        end if
    end for
    return S
end procedure
Require: T is a tree with multivariate nodes
procedure MBTLAYOUT(T)
    nodes ← CIRCULARTREEMAPLAYOUT(T)
    nodes ← RESORT(nodes)
    scene S ← FASTLAYOUT(nodes)
    update all circles in S by corresponding polar area diagrams
end procedure
```

7.1 Omics Dataset

Here, we evaluate our approaches on the omics data, which the Functree2 [10] visualization has focused on. The data consist of seven samples of NCBI gene identifiers [26] abundance, with the respective hierarchical relation being predefined in the KEGG database [17]. Layers of the tree correspond to functional categories, which from top to bottom are the biological category, biological process, and KEGG pathway. We compare our representations with Functree2 in two main analysis tasks: comparison and exploration (also demonstrated in the accompanying video).

Comparison. In Functree2, the comparison of a sample's value between nodes is difficult because the stacked bar charts are not aligned there. For this reason, interactive functionality is provided in Functree2, to select an attribute and map its values to circle areas for comparison between nodes, as shown in Figure 6a. Our MSB and MIP (Figure 6b–6e), in contrast, enable simultaneous comparison of all attributes in a single static representation. Although our representations have the strength of being able to compare multiple attributes, their weakness is with respect to the comparison of the total amounts of all attributes, which is the strength of Functree2. However, since such sum up is only applicable if all attributes have the same unit and could even be misleading (Simpson's paradox), we think this trade-off is definitely acceptable.

Exploration. In Functree2, if one wants, e.g., to understand how the values in the KEGG pathway relate to the values in upper layers, like the biological process or biological category, one can select a leaf node and examine the corresponding nodes in the upper layer with the help of the highlighted path (Figure 6a). On the other hand, localization of the dominant nodes in lower levels (e.g., locate nodes a, b, c in Figure 6a) in the static view of Functree2 without interaction is less easy. Because Functree2 is based on the radial tree, visual reading is complicated due to extended visual transitions [5]. Thus, such highlighting based on selection is very helpful when a tree has many levels. However, in Figure 6b, 6d, and 6e, finding such relations can be accomplished straightforwardly.

Figure 7 shows the same data using the MBT and the MCT. Figure 7a demonstrates the advantage of the MBT for visualizing properties of leaves over our other design variants. Figure 7b, on the other hand, shows how these multivariate quantities vary proportionally on the internal nodes of the MCT. From Figure 6b and Figure 7b, we can see the rather low compactness of the MCT.

7.2 National Statistics Dataset

Our next example is national statistical data [1], where all attributes differ in their units. The data consist of three layers: *Continent*, classified as Oceania, Africa, Asia, Europe, and America, *Subcontinent*, consisting of, e.g., Western Europe, and *Nation*. The mean operation (for derived quantities at internal nodes of the hierarchy) is applied to seven attributes, which are: surface area, population, GDP, sex ratio, urban population percentage, threatened species count, and CO_2 emission estimates. Since the units differ, we normalized each attribute separately in its range, leading to the results shown in Figure 8. In the MBT and the MCT as shown in Figure 9, we can clearly identify apparent multivariate features. Also, we can identify countries, e.g., the United States, exhibiting the largest GDP, large surface area, and large CO_2 emission estimate. Another example is China, having the largest population, large surface area, large GDP, and large CO_2 emission estimate. From Figure 8b, it can be seen that total CO_2 emission (violet) in East Asia is comparable to North America, but at the same time that population (red) of East Asia is much higher than that of North America. Only due to the integrated hierarchical multivariate visualization, one can directly identify from this the underlying principle, i.e., that the value of CO_2 emission per person is much lower in East Asia. One can also observe the multivariate features across different hierarchical levels, e.g., from Figure 8a and 8b. Although East Asia has a relatively high urban population percentage, due to the overall low development of Asia, this number is reduced at the *Continent* level.

8 User Study

So far, we have demonstrated with a case study that our four design solutions are useful for the analysis of hierarchical multivariate data. Here, we back this up with a user study, which evaluates our approaches with respect to the requirements that we determined in Section 3.3. For a more compact presentation, we introduce the following acronyms: MSB/MIP-S stands for MSB/MIP with the equal siblings strategy, whereas MSB/MIP-L stands for MSB/MIP with the equal leaves strategy.

8.1 Approach

Let R1–R5 refer to the five requirements we derived in Section 3.3. For R1 (static representation) and R2 (avoid occlusions), our four design solutions are static and 2D. We achieved these requirements. For R3 (simple interpretation) and R4 (readability), we conducted a user study online (due to the COVID-19 pandemic). The readability of hierarchical properties directly depends on the tree visualization technique, so it is not our main focus in the evaluation. In the user study, we mainly focus on the readability of the quantitative multivariate properties under the hierarchical layouts.

For R5 (arbitrary data on internal nodes), MSB/MIP-S and MSB/MIP-L achieved this requirement. For MBT, this seems to be very hard to achieve and would likely be hard to read. For MCT in case of not normalized attributes (not different units), only the proportion can be represented. In case of normalized attributes, the data can be represented on internal nodes.

Author's copy. To appear in Proceedings of 14th IEEE Pacific Visualization Symposium (PacificVis 2021).

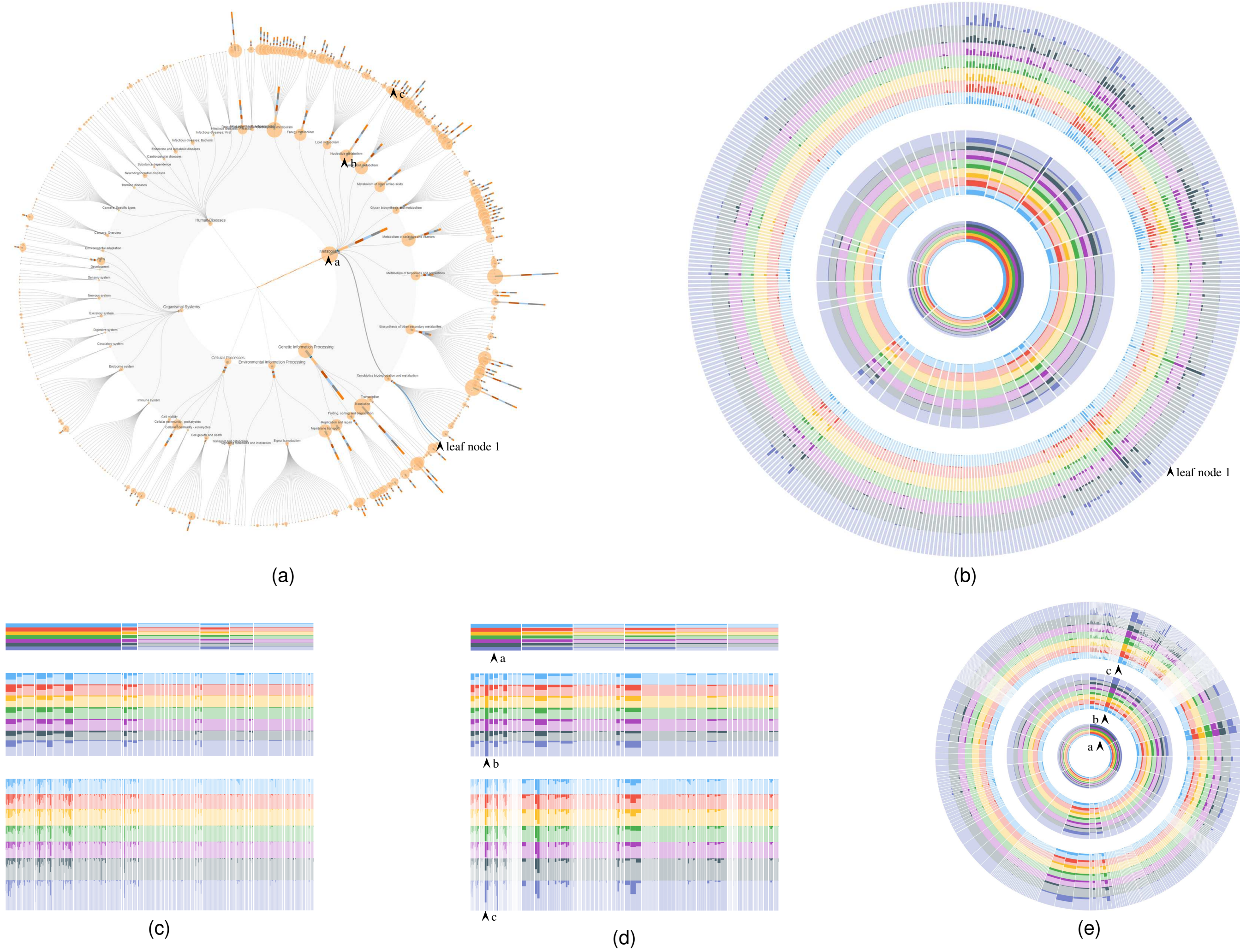

Figure 6. (a) Functree2, for comparison with our designs, ((b), (e)) MSB, and ((c), (d)) MIP, all demonstrated using the Omics dataset. Internal nodes show average of their children. Layout strategy with equal sibling (angular) width ((d), (e)), and with equal leaf (angular) width ((b), (c)).

8.2 Tasks

We derived eight tasks (T1–T8, with listed questions) from Requirement R4. R3 was evaluated indirectly by the correctness and completion time of T1–T8, and R4 was evaluated directly by T1–T8. For every representation, the same questions have been asked for simplifying the comparison. For the questions, the participant had three answers to choose from: A, B, or "hard to tell". A or B indicate a region, leaf, internal node, or a certain comparison according to the task. Task-relevant elements in the visualizations are indicated by arrows and labels.

- T1 – Comparison between two attributes in one internal node (R4): *Compare the length/height of the red and light blue region of internal node 1, which region is larger, A or B?*
- T2 – Comparison between two attributes in one leaf (R4): *Compare the area/height of the red and light blue region of leaf 1, which region is larger, A or B?*
- T3 – Comparison of one attribute between different leaves (R4): *Compare the area/height of the yellow region of leaf A and leaf B, which is larger?*
- T4 – Comparison of one attribute between a leaf and an internal node (R4): *Compare the length in its range of the yellow region of internal node A, and the area in its range of the yellow region of leaf B, which is larger? / Compare the height of the yellow region of internal node A and leaf B, which is larger?*
- T5 – Comparison of one attribute between internal nodes (R4): *Compare the length/height of the gray region of internal node A and internal node B, which is larger?*
- T6 – Estimation of the value of one attribute in a leaf (R4): *Estimate the range of the value of the light blue area (maximum is 1), A: <0.4 or B: >0.4?*
- T7 – Estimation of the value of one attribute in an internal node (R4): *Estimate the range of the value of the violet area (maximum is 1), A: <0.4 or B: >0.4?*
- T8 – Reading the hierarchy (R4): *The parent node of leaf 1 is internal node A or internal node B?*

8.3 Design

Participants. 22 participants were recruited from students and staff. 4 majored in mathematics, 11 in computer science, 3 in geoinformatics, 2 in bioinformatics, 1 in other natural science fields, and 1 in human science fields. 8 were female, 14 male.

Procedure. The participants were welcomed and asked to be focused during their participation. They were introduced to carry out the user study using full-screen display of their browsers and to not zoom in during the survey. In case they had no color weak-

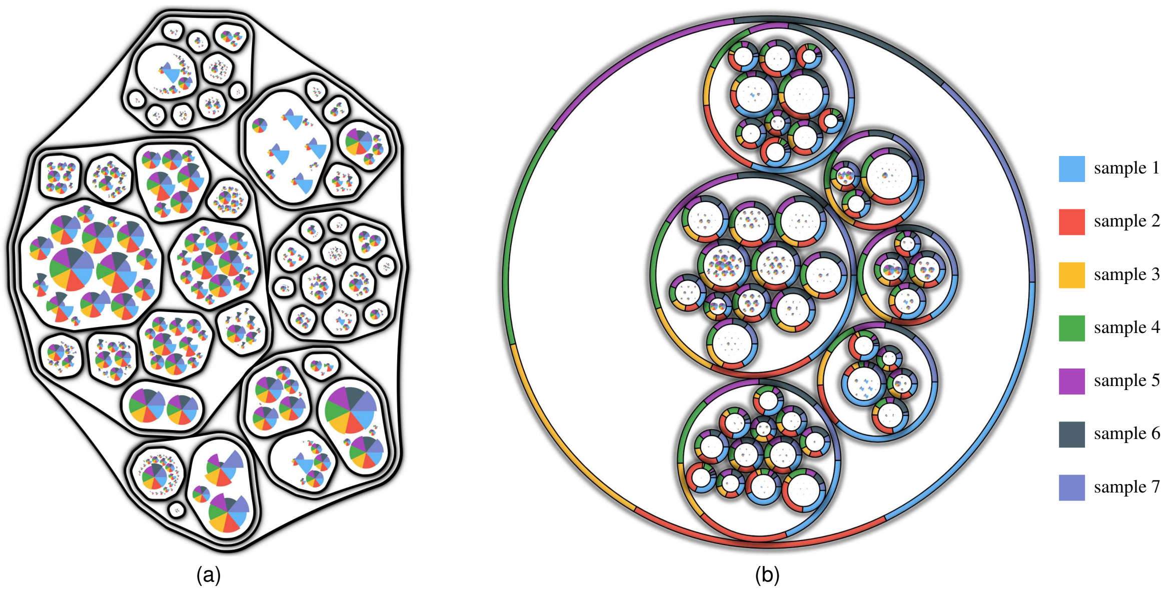

Figure 7. Omics dataset visualized with MBT (a), and MCT (b) with internal nodes showing the multivariate mean of their children.

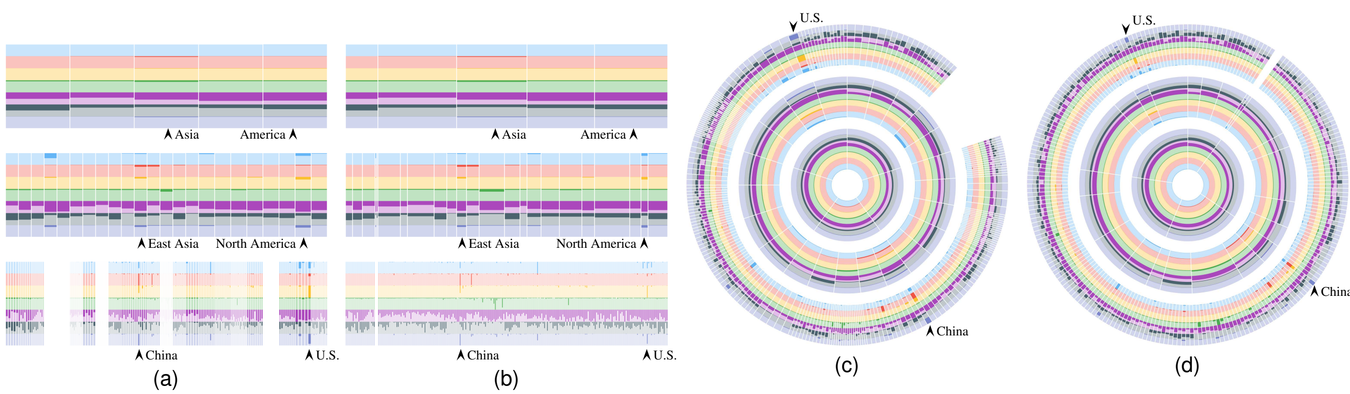

Figure 8. MSB ((c), (d)) and MIP ((a), (b)), demonstrated using National Statistics dataset. The color legend refers to Figure 9. Internal nodes show the average of their children. Layout strategy with equal sibling (angular) width ((a), (c)), and with equal leaf (angular) width ((b), (d)).

ness or color blindness, they were allowed to continue and were introduced to our four representations by an introduction video (see supplemental material). After they had understood the technique, all tasks, together with the representations, were presented to the participants one by one in a randomized order.

Data. We used the National Statistics dataset presented in Section 7.2. All pictures were presented at resolution 700×700 pixels.

Framework and measurements. We implemented the user study in the SurveyGizmo framework, which provided measurement of completion time of each task and the achieved accuracy.

8.4 Evaluation

The results of our user study are provided in Figure 10. We also performed an ANOVA test on the completion time of T1–T8 to analyze whether an actual difference exists between these representations. The results of F-value and p-value of T1–T8 are listed in Table 2.

Since T1, T4, and T7 have $p < 0.1$, the differences between these means are relatively statistically significant, which means it is reasonable to compare the means of completion time between these tasks. For other tasks with $p > 0.1$, we only compare the accuracy.

For T2, T3, and T6, which only concern leaves, MSB-L is one of the best candidates. Due to the low compactness of MCT (discussed in Section 5.2 and also observed in Section 7.1), it has the worst

Table 2. ANOVA test on the completion time of T1–T8.

Tasks	F-value	p-value
T1	F(4, 105) = 3.3588	p = 0.0125
T2	F(5, 126) = 0.4577	p = 0.8071
T3	F(5, 126) = 1.0871	p = 0.3708
T4	F(4, 105) = 2.0771	p = 0.0890
T5	F(4, 105) = 1.5499	p = 0.1932
T6	F(5, 126) = 1.2835	p = 0.2751
T7	F(4, 105) = 2.0117	p = 0.0981
T8	F(5, 126) = 0.4759	p = 0.7937

accuracy in these tasks. MBT, due to the compactness, in contrast, performed much better than MCT, comparable to MSB-L.

For T1, T5, and T7, which only concern internal nodes, in general, MSB and MIP do not exhibit significant differences. However, the equal sibling strategy is slightly better than the equal leaf strategy. MCT has the worst accuracy, because the stacked bar is curved along the circle packing line, which affects readability. MBT does not support such tasks.

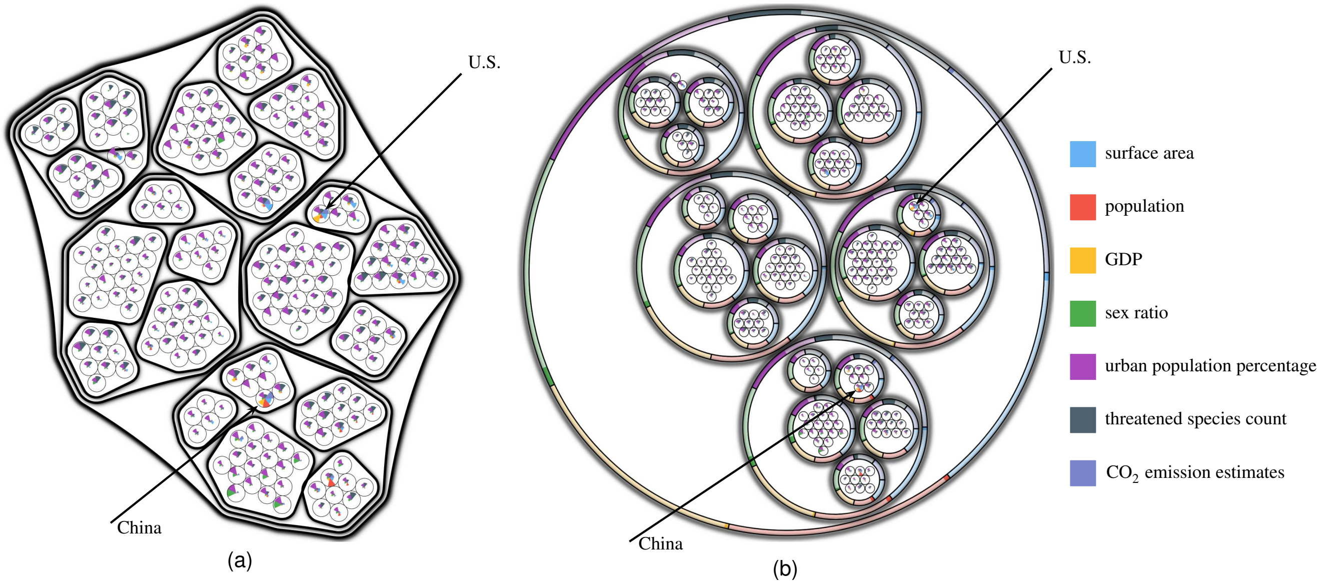

Figure 9. National Statistics dataset visualized with MBT (a), and MCT (b) with each internal node showing the multivariate mean of its children.

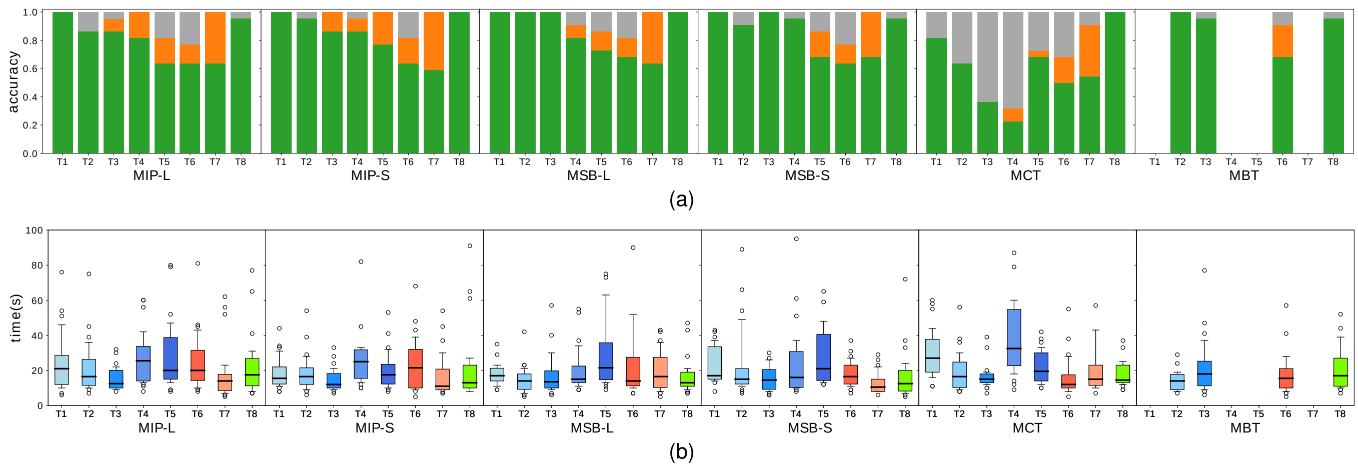

Figure 10. Accuracy (a) and completion time (b) for tasks T1–T8 and our different representations. (a) Correct answer (green), false answer (orange), and "hard to tell" answer (gray). (b) The whiskers represent the 10–90 percentile interval around the median, while the box represents the interquartile range (Q1–Q3).

For T4, MSB takes less time than MIP. Again, MCT has the worst performance. MCT performed the worst not only because of its weakness regarding presentation of leaves due to the low compactness, but also due to the difficulty of comparing a polar area in its range and a bar in its range. From the fact that MCT performed worst in almost all tasks, one can argue that Functree2 would perform badly in such a study due to its low compactness.

For T8, all representations performed quite well. As mentioned, the readability of the hierarchy mostly depends on the hierarchical visualization technique. Since the icicle plot, the sunburst chart, and the circular treemap are very widely used, and bubble treemaps as a variant to circular treemaps are also easy to understand, this result is in accordance with our expectations.

In general, we can see that the MSB is better than the MIP. We think this is because the sunburst chart has advantages regarding user acceptance compared to the icicle plot [33].

9 Limitations

We need to mention that the scalability of all our four representations is limited by the chosen tree/multivariate visualization components (Section 4.1 and 4.2). In practice, we suggest the number of nodes to be less than 500, and the number of attributes to be no more than 10. Further research on the scalability is still needed. Additionally, since our main focus is to "ease the comparisons", and considering the fact that the user's exploration and interpretation strongly depend on experience and are hard to evaluate, we focus our user study mainly on visual perception. Thus, further investigations on user's exploration and interpretation are needed. From the resulting representations, we see that the MBT is not able to depict values on internal nodes, and the MCT exhibits bad performance regarding visual perception due to its low compactness.

10 Conclusion

We presented a design study on the visualization of hierarchical multivariate data. First, we determined the respective requirements. This guided us in the selection of concepts from multivariate visualization and visual representation of hierarchical data. For the hierarchical component, this resulted in the sunburst chart, the icicle plot, the circular treemap, and the bubble treemap as the basic layout. For the multivariate aspect of the data, we found the stacked bar chart and the concept of the pie chart to be an appropriate basis for finding design variants that support integration with the hierarchical layouts. Finally, we determined four integrations for visualizing hierarchical multivariate data, with several subspaces of design adjustments. We demonstrated our solutions using different data, and evaluated them with respect to the previously determined requirements. Future work could include improving scalability and introducing interactive focus+context approaches. Additionally, in case the quantitative property of the multivariate component is not desired but the correlation between attributes is, replacing the polar area diagrams with chord diagrams could be further investigated.

Acknowledgments

We thank Zhao Yuan from DKFZ (German Cancer Research Center), other domain experts, all the participants of the user study, and the anonymous reviewers. This work has been supported by DFG (Deutsche Forschungsgemeinschaft) grant GSC 220 in the German Universities Excellence Initiative at Heidelberg University.

References

[1] UN database. https://data.un.org/.
[2] A. Arleo, W. Didimo, G. Liotta, and F. Montecchiani. GiViP: A visual profiler for distributed graph processing systems. In *Graph Drawing and Network Visualization*, pp. 256–271. Springer, 2018.
[3] L. Beaudoin, M. . Parent, and L. C. Vroomen. Cheops: A compact explorer for complex hierarchies. In *Proc. IEEE VIS*, pp. 87–92, 1996.
[4] R. D. Bergeron, W. Cody, W. Hibbard, D. T. Kao, K. D. Miceli, L. A. Treinish, and S. Walther. Database issues for data visualization: Developing a data model. In *Database Issues for Data Visualization*, pp. 1–15. Springer, 1994.
[5] T. Blascheck, K. Kurzhals, M. Raschke, M. Burch, D. Weiskopf, and T. Ertl. State-of-the-art of visualization for eye tracking data. In *Proc. EuroVis STARs*, 2014.
[6] R. Borgo, J. Kehrer, D. H. S. Chung, E. Maguire, R. S. Laramee, H. Hauser, M. Ward, and M. Chen. Glyph-based visualization: Foundations, design guidelines, techniques and applications. In *Proc. Eurographics STARs*, 2013.
[7] M. Bostock, V. Ogievetsky, and J. Heer. D3 data-driven documents. *IEEE Trans. Vis. Comput. Graph.*, 17(12):2301–2309, 2011.
[8] N. Bremer. A Closer Look at Labor. https://nbremer.github.io/occupationscanvas/.
[9] H. Chernoff. The use of faces to represent points in k-dimensional space graphically. *J. Am. Stat. Assoc.*, 68(342):361–368, 1973.
[10] Y. Darzi, Y. Yamate, and T. Yamada. Functree2: An interactive radial tree for functional hierarchies and omics data visualization. *Bioinformatics*, 35(21):4519–4521, 2019.
[11] D. Engel, R. Rosenbaum, B. Hamann, and H. Hagen. Structural decomposition trees. *Comput. Graph. Forum*, 30(3):921–930, 2011.
[12] F. Fischer, J. Fuchs, and F. Mansmann. Clockmap: Enhancing circular treemaps with temporal glyphs for time-series data. In *Proc. EuroVis Short Papers*, pp. 97–101, 2012.
[13] Y.-H. Fua, M. O. Ward, and E. A. Rundensteiner. Hierarchical parallel coordinates for exploration of large datasets. In *Proc. IEEE VIS*, pp. 43–50, 1999.
[14] J. Fuchs, F. Fischer, F. Mansmann, E. Bertini, and P. Isenberg. Evaluation of alternative glyph designs for time series data in a small multiple setting. In *Proc. CHI*, pp. 3237–3246, 2013.
[15] J. Görtler, C. Schulz, D. Weiskopf, and O. Deussen. Bubble treemaps for uncertainty visualization. *IEEE Trans. Vis. Comput. Graph.*, 24(1):719–728, 2018.
[16] X. He, Y. Tao, Q. Wang, and H. Lin. Multivariate spatial data visualization: A survey. *J. Vis. (Tokyo)*, 22(5):897–912, 2019.
[17] M. Kanehisa1, S. Goto, Y. Sato, M. Kawashima, M. Furumichi, and M. Tanabe. Data, information, knowledge and principle: Back to metabolism in KEGG. *Nucleic Acids Res.*, 42:D199–D205, 2014.
[18] D. A. Keim. Visual techniques for exploring databases. In *Proc. KDD*, 1997.
[19] D. A. Keim and H.-P. Kriegel. Visualization techniques for mining large databases: A comparison. *IEEE Trans. Knowl. Data Eng.*, 8(6):923–938, 1996.
[20] J. LeBlanc, M. O. Ward, and N. Wittels. Exploring n-dimensional databases. In *Proc. IEEE VIS*, pp. 230–237, 1990.
[21] I. Letunic and P. Bork. Interactive Tree Of Life (iTOL) v4: recent updates and new developments. *Nucleic Acids Res.*, 47(W1):W256–W259, 2019.
[22] L. K. Long, L. C. Hui, G. Y. Fook, and W. M. N. W. Zainon. A study on the effectiveness of tree-maps as tree visualization techniques. *Procedia Comput. Sci.*, 124:108–115, 2017.
[23] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM TOG*, 5(2):110–141, 1986.
[24] M. J. McGuffin and J.-M. Robert. Quantifying the space-efficiency of 2D graphical representations of trees. *Inf. Vis.*, 9(2):115–140, 2010.
[25] N. H. Müller, B. Liebold, D. Pietschmann, P. Ohler, and P. Rosenthal. Hierarchy visualization designs and their impact on perception and problem solving strategies. In *Proc. ACHI*, pp. 93–101, 2017.
[26] NCBI Resource Coordinators. Database resources of the national center for biotechnology information. *Nucleic Acids Res.*, 46(D1):D8–D13, 2018.
[27] C. Nobre, M. Meyer, M. Streit, and A. Lex. The state of the art in visualizing multivariate networks. *Comput. Graph. Forum*, 38(3):807–832, 2019.
[28] C. Nobre, M. Streit, and A. Lex. Juniper: A tree+table approach to multivariate graph visualization. *IEEE Trans. Vis. Comput. Graph.*, 25(1):544–554, 2019.
[29] R. M. Pickett and G. G. Grinstein. Iconographic displays for visualizing multidimensional data. In *Proc. IEEE SMC*, pp. 514–519, 1988.
[30] H.-J. Schulz. Treevis.net: A tree visualization reference. *IEEE Comput. Graph. Appl.*, 31(6):11–15, 2011.
[31] H.-J. Schulz, S. Hadlak, and H. Schumann. The design space of implicit hierarchy visualization: A survey. *IEEE Trans. Vis. Comput. Graph.*, 17(4):393–411, 2011.
[32] M. Sondag, W. Meulemans, C. Schulz, K. Verbeek, D. Weiskopf, and B. Speckmann. Uncertainty treemaps. In *Proc. IEEE PacificVis*, pp. 111–120, 2020.
[33] M. Sujay, R. Al Tarawneh, S. R. Humayoun, R. Moses, S. Panis, and A. Ebert. Radial vs. rectangular: Evaluating visualization layout impact on user task performance of hierarchical data. *IJCSIS*, 12:17–31, 2017.
[34] F. van Ham and J. J. van Wijk. Beamtrees: Compact visualization of large hierarchies. In *Proc. IEEE InfoVis*, pp. 93–100, 2002.
[35] J. J. Van Wijk and H. Van de Wetering. Cushion treemaps: Visualization of hierarchical information. In *Proc. IEEE InfoVis*, pp. 73–78, 1999.
[36] Z. Vosough, M. Hogräfer, L. A. Royer, R. Groh, and H.-J. Schulz. Parallel hierarchies: A visualization for cross-tabulating hierarchical categories. *Comput. Graph.*, 76:1–17, 2018.
[37] W. Wang, H. Wang, G. Dai, and H. Wang. Visualization of large hierarchical data by circle packing. In *Proc. CHI*, pp. 517–520, 2006.
[38] M. O. Ward. Multivariate data glyphs: Principles and practice. In *Handbook of Data Visualization*, pp. 179–198. Springer, 2008.
[39] R. Wettel and M. Lanza. Visualizing software systems as cities. In *Proc. IEEE VISSOFT*, pp. 92–99, 2007.
[40] K. Wittenburg and T.-Y. Lee. Equal-height treemaps for multivariate data. In *Proc. AVI Conference*, pp. 88:1–88:3, 2018.
[41] K. Wittenburg and T. Turchi. Treemaps and the visual comparison of hierarchical multi-attribute data. In *Proc. AVI Conference*, pp. 64–67. ACM, 2016.
[42] P. C. Wong and R. D. Bergeron. 30 years of multidimensional multivariate visualization. In *Scientific Visualization, Overviews, Methodologies, and Techniques*, pp. 3–33, 1997.
[43] L. Woodburn, Y. Yang, and K. Marriott. Interactive visualisation of hierarchical quantitative data: An evaluation. In *Proc. IEEE VIS Short Papers*, pp. 96–100, 2019.
[44] H. Zhao and L. Lu. Variational circular treemaps for interactive visualization of hierarchical data. In *Proc. IEEE PacificVis*, pp. 81–85, 2015.