

## Hierarchical Data Visualization for Atomic Plant Data\*

Takayuki ITOH\*\*, Shiho FURUYA\*\*, Hiroyuki OHSHIMA\*\*\* and Koji OKAMOTO†

\*\* Graduate School of Humanities and Sciences, Ochanomizu University  
2-1-1 Otsuka, Bunkyo-ku, Tokyo 112-8610, Japan  
E-mail: {itot, shiho}@itolab.is.ocha.ac.jp

\*\*\* Computational Fast Reactor Engineering Group, Japan Atomic Energy Agency  
4002 Narita, O-arai, Ibaraki 311-1393, Japan  
E-mail: ohshima.hiroyuki@jaea.go.jp

† Human and Engineered Environmental Studies, The University of Tokyo  
5-1-5 Kashiwanoha, Kashiwa-shi, Chiba 277-8563, Japan  
E-mail: okamoto@k.u-tokyo.ac.jp

### Abstract

This paper introduces our own hierarchical data visualization technique, HeiankyoView, and discusses its potential for fluid science. HeiankyoView represents hierarchy as nested rectangles, and displays thousands of data elements in one display space. The paper briefly introduces the algorithm and various applications of HeiankyoView, and discusses our attempts for visualization of computational fluid dynamics data. Finally, the paper introduces our work on visualization of atomic plant data using HeiankyoView.

**Key words** : Hierarchical data visualization, HeiankyoView, Computational fluid dynamics, Atomic plants.

### 1. Introduction

Many visualization techniques are categorized into two fields: scientific visualization and information visualization. Most of scientific visualization techniques represent various phenomena in physical spaces, which are usually obtained by real measurements or computational simulations. On the other hand, information visualization techniques represent general information, including non-scientific, non-geometric and non-spatial problems. These two fields have an independent history of research. However, some recent scientific visualization works have integrated information visualization techniques.

Shneiderman<sup>(1)</sup> reported that data structures for information visualization can be categorized into the following seven types: one-dimensional, two-dimensional, three-dimensional, more than three dimensional, tree, graph, and time-varying data. We have been interested in visualization of tree data ("hierarchical data" in this paper) for several years. Information in daily life is often hierarchically organized; examples include structures of human organizations and file systems of computers.

There have been many research projects on hierarchical data visualization. Many of these projects represent tree structures as a set of nodes and edges connecting parent and/or child nodes, where the concept is often called the "node-link approach". Many of these approaches provide user interfaces to explore the hierarchy interactively; Hyperbolic Tree<sup>(2)</sup> is one of the most famous works. Many of other works spread data items of the lowest hierarchy of the data, where the concept is often called a "space-filling approach"; TreeMaps<sup>(3)</sup> is one of the most famous works. We have also presented a space-filling hierarchical data visualization technique<sup>(4)(5)</sup>. The following sections briefly introduce our hierarchical data visualization technique and its applications.

## 2. Hierarchical Data Visualization Technique "HeiankyoView"

Figure 1(Left) shows an example of a tree, and Figure 1(Right) shows the representation of the same data by HeiankyoView, which represents leaf-nodes as painted square icons, and branch-nodes as nested rectangular borders.

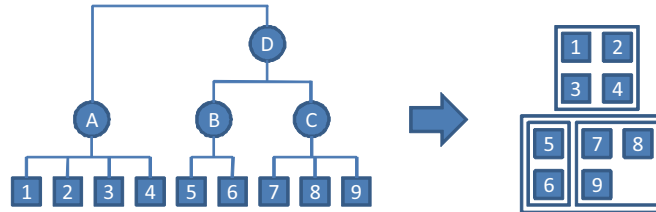


Fig. 1 (Left) Node-Link representation of a tree. (Right) Representation of the same data by HeiankyoView.

The processing flow of HeiankyoView is as follows. HeiankyoView first packs icons that represent the lowest-level leaf nodes (e.g., 1 to 4 in Figure 1) under a branch node (e.g., A in Figure 1), and then encloses the icons by rectangular borders, which correspond to the branch node. HeiankyoView packs a set of rectangular borders that belong to higher levels of the hierarchy, and generates the larger enclosing rectangular borders that correspond to the higher branch nodes (e.g., D in Figure 1). Repeating the process from the lowest level of the hierarchy toward the highest level, HeiankyoView places all of the data items onto the layout area. Here, the packing algorithm for icons and rectangles is the key technology for HeiankyoView. We presented a triangular-mesh-based rectangle packing algorithm<sup>(6)</sup>, and then applied an improved rectangle packing algorithm, as described in<sup>(4)(5)</sup>. Both algorithms place icons and rectangles one-by-one onto display spaces, while the algorithms choose their positions from multiple candidate positions.

HeiankyoView places thousands of leaf-nodes into one display space while satisfying the following conditions:

- It never overlaps the leaf-nodes and branch-nodes in a single hierarchy of other nodes.
- It attempts to minimize the display area requirement.
- It draws all leaf-nodes with equally shaped and sized icons.

This representation style is suitable for visualizing thousands of leaf-nodes of hierarchical data in one display space, rather than representing hierarchical relationships between parent and children nodes. All of the leaf-nodes are displayed as equally-sized clickable icons.

The naming of HeiankyoView came from "Heiankyo", an ancient palace in Kyoto-city. Since all blocks in Heiankyo orthogonally align, and therefore the visualization results of HeiankyoView often look somewhat similar to the map of Heiankyo. That is a reason why we named this technique HeiankyoView.

## 3. Extensions and Applications of HeiankyoView

We have applied HeiankyoView to a variety of hierarchical data.

Our early works focused on visualization of statistics of computational events recorded in various log files. We first visualized the statistics of Web accesses, which forms a hierarchy of Web pages according to their directory structures, and represents the numbers of accesses of each Web page by colors or heights of icons. We also visualized network intrusion detection results<sup>(4)</sup>, which forms a hierarchy of computers according to their IP addresses, and represents the number of malicious accesses of each computer by the heights of icons. These works aimed to provide all-in-one display of statistics of accesses, allowing users to discover correlations between hierarchy structures and the accesses. As a result, we discovered concentrated accesses to specific organizations, based on the visualization results.

Recent our works focus on the visualization of clustered data. We visualized the experimental values of drugs, which are clustered according to their chemical structures<sup>(5)</sup>. We also

visualized the clustering results of questionnaires, newspaper articles, gene expression data, and partial geometric features of protein molecular surfaces.

Figure 2 shows an example of visualization of newspaper articles, which displays two HeiankyoView visualizations from side to side. Here, the left side represents clustering result of keywords, and the right side represents clustering result of documents. We call this technique "Sakyo and Ukyo", where "Sakyo" is the east side of Heiankyo, and "Ukyo" is the west side of Heiankyo. HeiankyoView provides the user interface so that the user can interact with the both the left and right sides at the same time. When the user clicks icons corresponding to keywords in the left side, the importance of the keywords in the documents is indicated by the colors in the right side. Similarly when clicking icons corresponding to documents in the right side, the body of the document is indicated in another window, and sum of the importance for the cluster of clicked documents is indicated by colors in the left side.

Figure 3 shows an example of visualization of hierarchical drug data<sup>(7)</sup>. Here, five colors represent susceptibility to five isozymes (CYP1A2, CYP2C9, CYP2C19, CYP2D6, and CYP3A4) for each drug. We call this technique "JunihitoeView", where "Junihitoe" is a colorful costume in Heian Period. This technique is suitable for the visualization of multi-dimensional (5-dimensional in this example) hierarchical data.

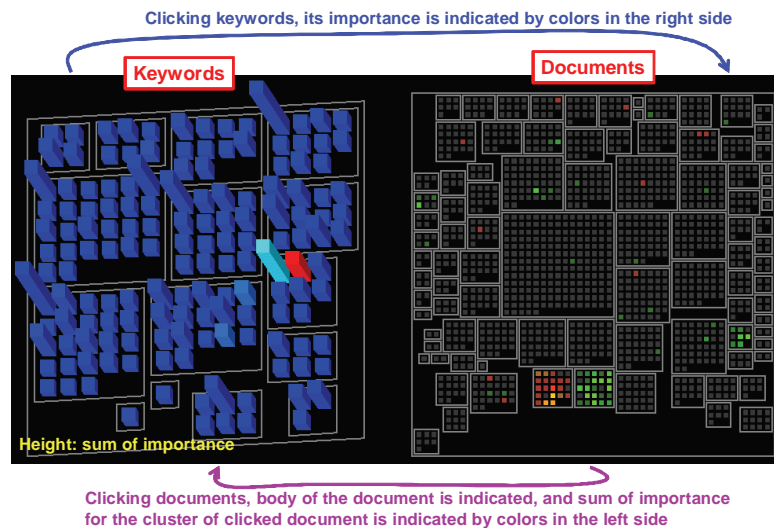


Fig. 2 "Sakyo and Ukyo" for visualization of corpus data.

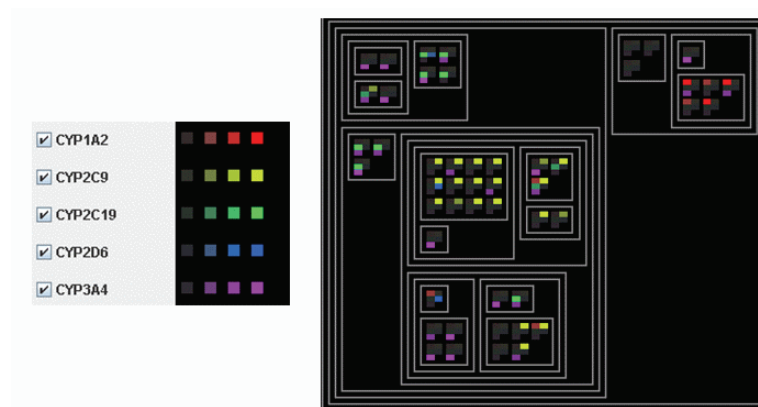


Fig. 3 "JunihitoeView" for visualization of drug data.

We are also using HeiankyoView as a content browser. As shown in Figure 4, we presented "CAT" (Clustered Album Thumbnails), a browser of large-scale image collections<sup>(8)</sup>, which maps many images onto icons. We also presented a similar work for visualization of

music collections, which displays automatically selected music icon images.



Fig. 4 "CAT" for visualization of an image collection.

In summary, most of the presented applications of HeiankyoView deal with non-geometric and non-spatial information.

#### 4. Visualization of Fluid Dynamics Simulation Results by HeiankyoView

This section considers the visualization of fluid dynamics simulation results in HeiankyoView. Figure 5(Left) shows an example of a visualization of a weather simulation result. This example is from volume data consisting of orthogonally aligned grid points and rectangular elements. Each grid-point has two scalar values (pressure and temperature) and a vector value (air flow). The figure represents distribution of the pressure by three colors of isosurfaces, and the air flow by several streamlines.

Figure 5(Right) shows an example of visualization of the same volume data by HeiankyoView. To obtain the example, we first extracted 16 slice planes from the volume along the x-axis, and selected 9x9 grid points from the each of the slice planes. We visualized the data as 16 groups of 81 bars, by calculating their colors according to the pressure, and their heights according to the temperature.

Here, both examples denote high pressure by red, and low pressure by blue or green. We can recognize that the pressure varies mainly along the z-axis, not the x- or y-axis, from the left example. We can also recognize that the pressure does not vary so much along x-axis from the right example compared with the left example. Meanwhile, we can easily recognize the coherency between pressure and temperature from the right figure. Briefly, temperature gets higher when pressure is lower.

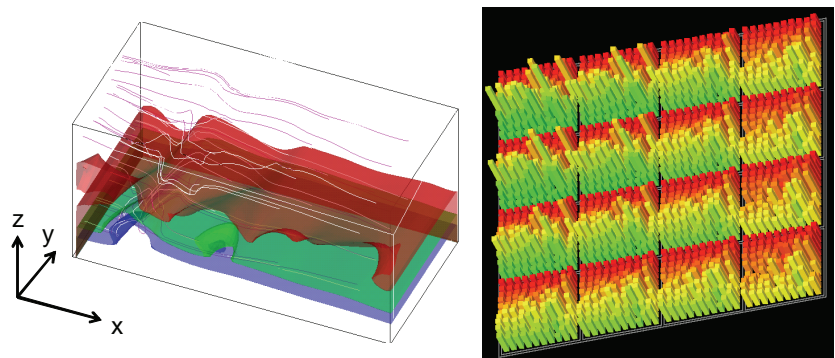


Fig. 5 (Left) Visualization of volume data by isosurfaces and streamlines. (Right) Visualization of volume data by HeiankyoView.

We think that most of researchers and engineers working with fluid dynamics simulation used to rely on volume visualization as shown in Figure 5(Left) rather than Figure 5(Right).

One reason for this preference may be that HeiankyoView is not very suitable to visualize the spatial coherency of scalar fields. To solve the problem we may need to provide the linkage to physical space visualization techniques. Providing the user interface to interact the left and right visualization results in Figure 5, as Sakyo and Ukyo provides as shown in Figure 2, will be more comprehensive.

On the other hand, we think that information visualization techniques may provide different useful aspects to visualize volume data. Here we discuss the features of visualization of fluid dynamics data by HeiankyoView.

Simultaneous representation of multi-dimensional values is a feature of information visualization techniques. Most volume visualization techniques represent distribution of scalar values by colors; particularly in Figure 5(Left), colors of isosurfaces denote the scalar values. Here, HeiankyoView can be better in visualizing multi-dimensional values rather than such isosurface representations. Figure 5(Right) shows an example in which HeiankyoView represents two scalar fields simultaneously, by colors and heights of the bars. The representation is suitable to understand the distribution and correlation of the two scalar fields. We experimentally determine to assign scalar fields as follows:

- We assign a scalar field to heights, if larger values are more important or caution needed.
- Otherwise, we assign the scalar field to colors.

If we do not have any scalar fields suitable for heights, or if we would like to simultaneously visualize more than two scalar fields, JunihitoeView will be more effective.

Another feature is a discovery of very local but important phenomena. Indirect volume visualization techniques (e.g., isosurface, streamline) represent partial features of scalar or vector fields; therefore, indirect visualization may miss important portions of the fields. Direct volume visualization techniques (e.g., volume rendering) represent the whole distribution of scalar or vector fields by using semi-transparent rendering techniques. It is a good approach to briefly look the overall distribution of the fields; however, if transfer functions or other configurations are not finely adjusted, small features will not be noticeable. Automatic transfer function setting has been an area of active research regarding solving this problem<sup>(9)</sup>. HeiankyoView can be an alternative approach to represent very local but important features of scalar fields.

## 5. Visualization of Measurements and Simulations of Atomic Plants

This section introduces our visualization work of measurements and simulations of atomic plants, applying our hierarchical and time-varying data visualization techniques.

### 5.1. Visualization of Atomic Plant Data by HeiankyoView

We visualized actual measurement data from Monju, a prototype fast breeder reactor built in Fukui prefecture. Monju has a measurement recording system, called MIDAS, which stores 512 independent measurements, including temperatures, pressures, velocities, voltages, and so on. We constructed a hierarchy of the 512 points by two-level division. First we divided them according to their positions, and then according to their units (e.g., temperature, pressure, and velocity). Then, we visualized the hierarchical data by HeiankyoView, where spatially closer and identical unitary measurement values become concentrated in one rectangular area. We represented the measurement values by the colors and heights of the bars.

Figures 6 and 7 denote transfer functions to calculate the colors and heights of the bars. Our implementation assumes that lower and upper limits of steady state are set for each measurement. If the measured value is within the range of lower and upper limits, HeiankyoView draws the bar corresponding to the measured value as gray, because the value does not require the user's attention. If the measured value is higher than the upper limit, HeiankyoView draws the bar as red or yellow. If the measured value is lower than the lower limit, HeiankyoView draws as blue or green. Our implementation also assumes that difference of measured values

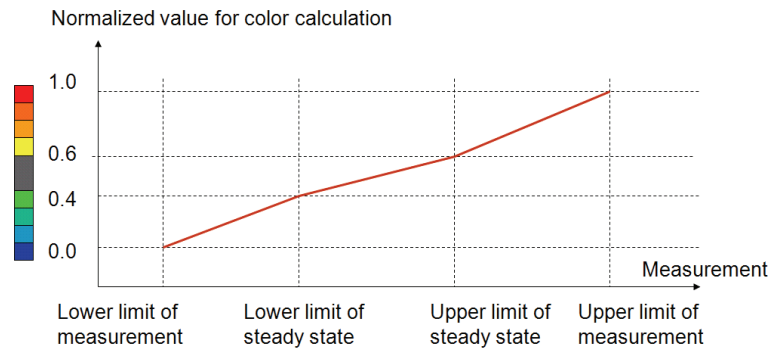


Fig. 6 Color transfer function.

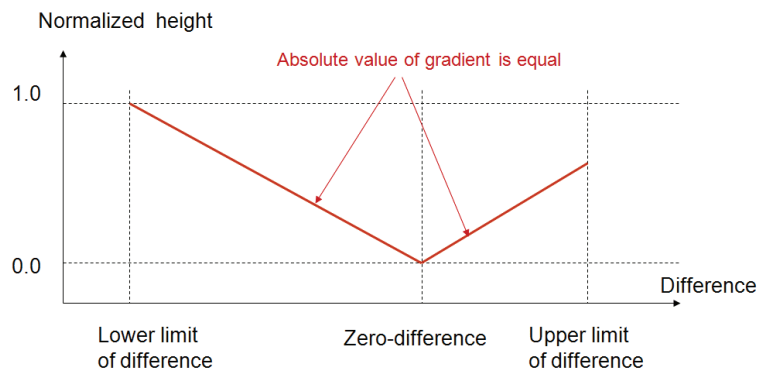


Fig. 7 Height transfer function.

between current and previous time can always be calculated. If the absolute value of the difference is large, HeiankyoView draws a tall bar corresponding to the measured value, because the value requires the user's attention. Otherwise, HeiankyoView draws the bar as short.

Figure 8 shows a visualization result, where a bar in a pink circle denotes that the measured value was suddenly increasing. At that time Monju had an anomaly: liquid sodium leaked from a pipe of a cooling system, and therefore the temperature readings suddenly jumped very high. The sudden change was observed at only one point, but this change certainly warranted operator attention. HeiankyoView can place emphasis on the visualization of important points, distinguishing them graphically from the large number of uninteresting points.

Figure 9 shows another visualization result, where many blue short bars denote that there were many measured values that were extremely low but staying flat. Several minutes before that time, Monju had a trip operation to stop the system, and then many instruments reported extremely high or low values. The blue short bars suggest that the trip operation proceeded safely, because the measured values were extremely low but stable. Such bars intensively appear in several specific groups, as shown in pink circles in Figure 9. These graphical elements denote that the trip operation works normally. Physical values in the trip operation go down block-by-block, and this phenomenon is well-shown in the visualization result.

In addition to the visualization of real measurement data shown in Figures 8 and 9, we are working with simulated data of atomic plants. We think a similar approach can be applied to various fluid science problems as well as to measurements and simulations of atomic plant data.

## 5.2. Visualization of Time-varying Atomic Plant Data

We assume that measurement or simulation data of atomic plants is collectible into tables, where columns denote measurement points, and rows denote the time. Here, examples



Fig. 8 Visualization result (1). A bar in a pink circle denotes that the measured value is suddenly increasing.

of visualization of atomic plant data in Figures 8 and 9 represent sets of hierarchical measurement values at specific times. On the other hand, we think that many users are interested in the visualization of time variation of the measurement values, in addition to the hierarchical visualization at a specific time. Let us discuss the simultaneous visualization of time-varying and hierarchical data in one display.

Figure 10 shows a snapshot of the presented technique, which displays two windows. The left thin window represents time-varying data, and the right window represents hierarchical data. The left window contains many buttons that correspond to specific times. When a user presses one of the buttons, the right window represents the hierarchical measurement values at the specified time.

Here we assume that the width of the black drawing area of the thin window is approximately 150 or 200 pixels. It is not always easy to visualize time-varying data of the hundreds or thousands of measurement values in such a narrow drawing area. To solve the problem, we applied a concept of a quantization table, which is often used in compression of sound data, for the visualization of large-scale time-varying data in the narrow drawing area.

Sound compression techniques look up the quantization table to multiply the coefficients to each frequency elements, so that important frequency signals are finely coded, and other frequency signals are roughly coded. Our technique is inspired by the above mechanism: the technique finely represents remarkable measurement values, such as the values that are considerably higher or lower than usual values, or the values which are drastically increasing or decreasing. The technique realizes importance-based representation of the time-varying data.

Let  $n$  be the number of measurement values, and  $m$  be the horizontal resolution of the drawing area. Also, take  $f_i(t)$  as  $i$ -th measurement value at the time  $t$ , where  $f_i(t)$  is normalized within the range  $[0, 1]$ . The technique first calculates the importance of the  $i$ -th measurement value  $p_i$ . Here, calculation of importance is application oriented, but in the particular case of atomic plant data we calculate  $p_i$  by the following equation:

$$p_i = S_1|f_i(t) - 0.5| + S_2|f'_i(t)| \quad (1)$$

where  $S_1$  and  $S_2$  are positive constant values.

The technique then calculates the width of the  $i$ -th measurement value,  $w_i$ , by the follow-



Fig. 9 Visualization result (2). Many blue short bars denote that there are many measured values that are extremely low but staying flat.

ing equation:

$$w_i = m \frac{p_i}{\sum_{i=1}^n p_i} \quad (2)$$

After calculating the widths of all measurement values by the above equation, the technique represents the distribution of the measurement values at  $t$ . The technique sorts the measurement values according to  $f_i(t)$ , and calculates colors of the measurement values. Finally, it paints rectangular regions corresponding to the measurement values, in the sorted order, applying the calculated colors and the widths  $w_i$ .

Figure 11 shows examples of the time-varying data. A circle drawn in Figure 11(Left) denotes the time shown in Figure 8. At the time only one measurement value drastically increased, and it is emphatically represented in Figure 11(Left). A circle drawn in Figure 11(Right) denotes the time shown in Figure 9. At the time many measurement values were extremely low, and they are certainly represented in Figure 11(Right).

We call our time-varying data visualization technique "KonjakuMonogatari", coming from the name of a novel in Heian Period which reflects the old stories.

One possible problem with KonjakuMonogatari is that it may cause time-space aliasing. If a user randomly chooses times by pressing the buttons, HeiankyoView will suddenly updates the display of values. It may be difficult to recognize the changes of values properly. We would like to implement a morphing technique between two specified times, so that the technique can smoothly represent the changes of values and make it easier to recognize them.

We had a user experience of the presented technique with 11 examinees. In this experience we asked the examinees to check caution needed times by visualizing the results of seven atomic plant simulations. We also asked a specialist of atomic plant simulation to similarly check caution needed times by visualizing the same data. The result was successful, since the number of examinees who checked the times reported by the specialist was 9.5 in average.

There have been several related works in the visualization of time-varying values using thin drawing areas. ThemeRiver<sup>(10)</sup> represents time-varying frequency values of keywords in large scale documents, as a metaphor of a river in thin drawing areas. It looks very similar to our visualization results, but it did not focus on visualization of distribution of hundreds or thousands of time-varying values. T-Map<sup>(11)</sup> represents the change of topological features of scalar fields in thin drawing areas. T-Map is also different from our technique, since it did not focus on visualization of distribution of hundreds or thousands of time-varying values themselves.



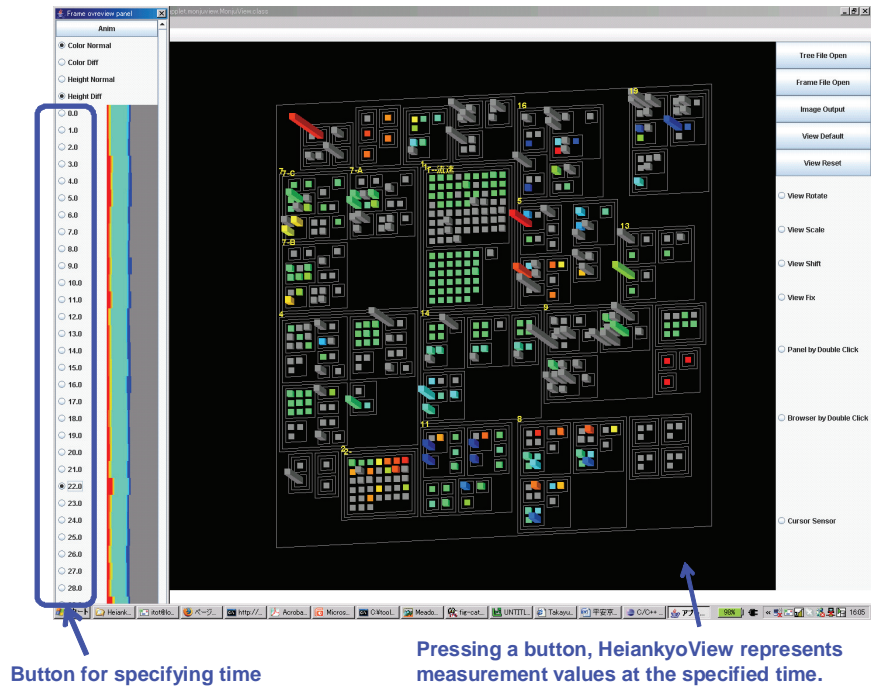


Fig. 10 Visualization of time-varying data on a thin window.

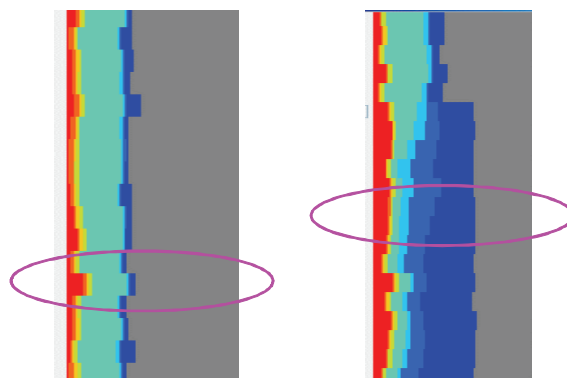


Fig. 11 Visualization result (3). (Left) Time-varying data around the specific time shown in Figure 8. (Right) Time-varying data around the specific time shown in Figure 9.

## 6. Conclusion

This paper first introduced our hierarchical data visualization technique HeiankyoView and its applications. The paper then discussed the visualization of fluid dynamics simulation results by HeiankyoView, and introduced our work on the visualization of measurements and simulations of atomic plants.

We think that information visualization techniques (including HeiankyoView) have potential applications to fluid science, and would like to demonstrate this potential in our following our future work:

- Experiments visualizing various fluid dynamics simulation data using HeiankyoView, in addition to the visualization of simulation data of atomic plants.
- Development of information visualization techniques for other types of data structures (e.g., time-varying or multi-dimensional data), for visualizing fluid dynamics simulation data.
- Integration of volume visualization techniques and HeiankyoView (or other information visualization techniques), for more intelligent analytics and discovery.

## Acknowledgements

Ms. Fumi Yasuda and Prof. Tetsuya Kawamura, with Ochanomizu University, provided volume data of weather simulation. Mr. Takaaki Sakai and Mr. Takashi Iwasaki, with Japan Atomic Energy Agency, supported the arrangement of measurement and simulation data of atomic plants.

The present study includes the results of "Large-scale information visualization for management of atomic systems" entrusted to "Ochanomizu University" by the Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT).

## References

- (1) Shneiderman, B., The Eyes Have It: A Task by Data Type Taxonomy for Information Visualization, *Proceedings of IEEE Symposium on Visual Languages '96* (1996), 336-343.
- (2) Lamping, J., Rao, R., The Hyperbolic Browser: A Focus+context Technique for Visualizing Large Hierarchies, *Journal of Visual Languages and Computing*, Vol. 7, No. 1 (1996), 33-55.
- (3) Johnson, B., Shneiderman, B., Tree-Maps: A Space Filling Approach to the Visualization of Hierarchical Information Space, *Proceedings of IEEE Visualization '91* (1991), 275-282.
- (4) Itoh, T., Takakura, H., Sawada, A., Koyamada, K., Hierarchical Visualization of Network Intrusion Detection Data in the IP Address Space, *IEEE Computer Graphics and Applications*, Vol. 26, No. 2 (2006), 40-47.
- (5) Itoh, T., Yamashita, F., Visualization of Multi-dimensional Data of Bioactive Chemicals Using a Hierarchical Data Visualization Technique "HeiankyoView", *Proceedings of Asia Pacific Symposium on Information Visualization (APVIS) 2006* (2006), 23-29.
- (6) Itoh, T., Yamaguchi, Y., Ikehata, Y., Kajinaga, Y., Hierarchical Data Visualization Using a Fast Rectangle-Packing Algorithm, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 10, No. 3 (2004), 302-313.
- (7) Yamazawa, M., Itoh, T., Yamashita, F., Visualization and Level-of-Detail Control for Multi-Dimensional Bioactive Chemical Data, *Proceedings of 12th International Conference on Information Visualization (IV08)* (2008).
- (8) Gomi, A., Miyazaki, R., Itoh, T., Li, J., CAT: A Hierarchical Image Browser Using a Rectangle Packing Technique, *Proceedings of 12th International Conference on Information Visualization (IV08)* (2008).
- (9) Takahashi, S., Takeshima, Y., Fujishiro, I., Topological Volume Skeletonization and Its Application to Transfer Function Design, *Graphical Models*, Vol. 66, No. 1 (2004), 24-49.
- (10) Havre, S., Hetzler, B., Whitney, P., Nowell, L., ThemeRiver: Visualizing Thematic Changes in Large Document Collections, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 8, No. 1 (2002), 9-20.
- (11) Fujishiro, I., Otsuka, R., Takahashi, S., Takeshima, Y., T-Map: A Topological Approach to Visual Exploration of Time-Varying Volume Data, *Springer Lecture Notes in Computer Science*, Vol. 4759 (2008), 176-190.