Visualizing Information on a Sphere

M. H. Gross, T. C. Sprenger, J. Finger

We describe a method for the visualization of information units on spherical domains which is employed in the banking industry for risk analysis, stock prediction and other tasks. The system is based on a quantification of the similarity of related objects that governs the parameters of a mass-spring system. Unlike existing approaches we initialize all information units onto the inner surface of two concentric spheres and attach them with springs to the outer sphere. Since the spring stiffnesses correspond to the computed similarity measures, the system converges into an energy minimum which reveals multidimensional relations and adjacencies in terms of spatial neighborhoods. Depending on the application scenario our approach supports different topological arrangements of related objects. In order to cope with large data sets we propose a blobby clustering mechanism that enables encapsulation of similar objects by implicit shapes. In addition, we implemented various interaction techniques allowing semantic analysis of the underlying data sets. Our prototype system IVORY is written in JAVA, and its versatility is illustrated by an example from financial service providers.

Visualizing Information on a Sphere

M. H. Gross, T. C. Sprenger, J. Finger

We describe a method for the visualization of information units on spherical domains which is employed in the banking industry for risk analysis, stock prediction and other tasks. The system is based on a quantification of the similarity of related objects that governs the parameters of a mass-spring system. Unlike existing approaches we initialize all information units onto the inner surface of two concentric spheres and attach them with springs to the outer sphere. Since the spring stiffnesses correspond to the computed similarity measures, the system converges into an energy minimum which reveals multidimensional relations and adjacencies in terms of spatial neighborhoods. Depending on the application scenario our approach supports different topological arrangements of related objects. In order to cope with large data sets we propose a blobby clustering mechanism that enables encapsulation of similar objects by implicit shapes. In addition, we implemented various interaction techniques allowing semantic analysis of the underlying data sets. Our prototype system IVORY is written in JAVA, and its versatility is illustrated by an example from financial service providers.

Visualizing Information on a Sphere

M. H. Gross, T. C. Sprenger, J. Finger[*]

Department of Computer Science, Federal Institute of Technology, Zürich, Switzerland
[*]Department of Computer Science, University of Freiburg, Germany

ABSTRACT

We describe a method for the visualization of information units on spherical domains which is employed in the banking industry for risk analysis, stock prediction and other tasks. The system is based on a quantification of the similarity of related objects that governs the parameters of a mass-spring system. Unlike existing approaches we initialize all information units onto the inner surface of two concentric spheres and attach them with springs to the outer sphere. Since the spring stiffnesses correspond to the computed similarity measures, the system converges into an energy minimum which reveals multidimensional relations and adjacencies in terms of spatial neighborhoods. Depending on the application scenario our approach supports different topological arrangements of related objects. In order to cope with large data sets we propose a blobby clustering mechanism that enables encapsulation of similar objects by implicit shapes. In addition, we implemented various interaction techniques allowing semantic analysis of the underlying data sets. Our prototype system IVORY is written in JAVA, and its versatility is illustrated by an example from financial service providers.

Keywords: information visualization, physically-based systems, multidimensional information space, hierarchies, blobby clustering

1 INTRODUCTION

The visualization of complex, multidimensional, non-numeric information and of their relationships is an emerging subfield of increasing importance in scientific visualization. Nowadays, global computer networks and distributed data bases, such as the world wide web (WWW), provide platforms for new dimensions of retrieval systems for information units. As a consequence, the scientific visualization and computer graphics communities have been challenged to develop advanced tools for understanding, navigating and interactively analyzing the associated information spaces. However, as opposed to most of the classical data sets in scientific visualization, information spaces carry over entirely new qualities of problems. The most important ones can be summarized as follows:

- *Multidimensional relationships:* information units are generally related to many other units. The resulting topological organization corresponds to a multidimensional graph. Thus, adjacencies cannot be visualized straightforwardly and have to be mapped into subspaces. Here, we can carry over some interesting methods already being used in graph layout.
- *Measuring similarity:* In contrast to many scientific data sets, information space is an abstract entity and there is no specific reason to employ Euclidean metrics to project similarities into a 3 dimensional subspace. Moreover, up to now, there is no mathematical framework or paradigm, on how to map scores and similarities provided by retrieval systems onto a model in a three-dimensional world.
- *Clustering and hierarchies:* The huge amount of information forces the use of a multiresolution setup. Hence, appropriate methods for the clustering of objects and for interactive level-of-detail control are needed.

Due to the importance of information visualization for many applications, various interesting approaches can be found in the literature and excellent surveys are available [15]. [4] and [10] for instance, visualized text documents and clusters as galaxies and themescapes, whereas [3] proposed cone trees which specifically address hierarchical organization. Another promising method is [8] or [12], who essentially used self-organizing schemes and neural networks to arrange information objects of the WWW. In a more general understanding, multidimensional visualization problems have been stressed in [5] or [1]. Here, mathematical projection algorithms were introduced to map data into subspaces while preserving their most important features. Interestingly, many current methods use physically-based paradigms, such as [14] or [9], where information units are taken as nodes of some generalized mass spring system revealing the structure of relations upon relaxation. These types of multidimensional visualization methods have been studied extensively in graph theory, and efficient algorithms had been introduced for fast graph relaxation, such as [8] and [13].

The work reported in our paper was inspired by the research summarized above. However, unlike existing methods, our approach is mostly application driven, where the context was focussed on visualization problems arising in financial service providing. Therefore, we defined a proprietary mathematical framework for quantifying relationships in information space. In addition, we propose a visualization paradigm that considers all information units initially located on the inner part of two concentric spheres and attached with springs to the outer one. The strengths of relations of different objects are correlated to the stiffnesses of springs between them. After initialization, the system converges into an equilibrium stage by solving the underlying differential equations using popular strategies [2], [11]. Thus, the energy minimum represents spatial adjacencies of objects which are similar to each other in information space. Moreover, since the similarity matrix also encodes the topology of the underlying graph, standard algorithms from graph theory can be employed to discover the indirect links of objects and to find minimal paths between them. This paper extends and further develops the approaches being reported in [16]. A major novelty is the introduction of a blobby clustering algorithm that helps to simplify the geometric complexity of large scale data sets. Therefore, a blob function is assigned to each spatial object position and the clustering hull is obtained from an implicit function rendered by an isosurface algorithm [21]. The method is simple, fast and shows up against spherical [6] or principal component analysis (PCA) based clustering [16] proposed earlier.

The organization of the paper is as follows: First, for reasons of readability, we briefly sketch our metric for similarity in information space. Section 3 introduces the paradigm of two concentric spheres and describes some strategies for initial positioning and clustering of information units. In section 4 interactive blobby clustering is elucidated. Finally, we illustrate the performance of our method by some examples related to data analysis tasks in private and public banking.

2 SIMILARITY IN INFORMATION SPACE

This section introduces some foundations required to understand the approach. First, our metric for similarity in information space is elaborated, since it represents a major prerequisite for visualization. Due to the rich literature on that subject we assume the reader to be familiar with the principles of mass-spring and particle systems. Efficient implementations can be found in [10] or [8].

One of the very challenging problems of information visualization is the definition of a mathematical framework for the quantification of similarity of entities in information space. We decided to found our framework on Euclidean vector spaces, but it also supports probabilistic and Boolean approaches for information retrieval. Specifically, we assume the metric to be computed in a preprocessing step, and providing the input parameters of a physically-based system.

Recalling some foundations of data base research we formalize the information retrieval process as follows: Let $A=\{A_1, .., A_k\}$ be a set of attributes (key words) and consider n objects $O=\{O_1, .., O_n\}$, where each object O_i is assigned to a score vector $\mathbf{s}_i$ of dimension k. The component s_{il} represents the relative importance of document O_i with respect to key word A_l and is assumed to be bounded by [0, 1].

We now define the so-called *similarity* c_{ij} of two objects O_i and O_j by the dot product ($\bullet$) of the associated scores normalized with respect to the dimension k:

$$c_{ij} = c_{ji} = \sqrt{\frac{\mathbf{s}_i \bullet \mathbf{s}_j}{k}} = \sqrt{\frac{1}{k}\sum_{l=1}^{k} s_{il}s_{jl}} \quad i,j = 1...n \qquad (1)$$

The resulting elements $0 \leq c_{ij} \leq 1$ form a symmetric, positive definite $n \times n$ *similarity matrix* $\mathbf{C}$:

$$\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & ... & c_{1n} \\ c_{21} & c_{22} & ... & c_{2n} \\ ... & ... & ... & ... \\ c_{n1} & ... & ... & c_{nn} \end{bmatrix} \qquad (2)$$

where $c_{ij} = c_{ji}, 1 \leq i, j \leq n$.

The *self-similarity* c_{ii} of an object O_i is provided by the normalized length of the score vector

$$c_{ii} = \frac{|\mathbf{s}_i|}{\sqrt{k}}, i = 1...n \qquad (3)$$

Note that the similarity matrix can be considered a float valued extension of the adjacency matrix which is commonplace in graph theory. For reasons of brevity, further details, such as normalization or graph topologies, are skipped and the reader is referred to [16] for detailed elaborations.

3 INFORMATION ON A SPHERE

Conforming to earlier approaches in information visualization and graph layout our paradigm combines the metric introduced above with the mass-spring approach. More precisely, we assign the computed similarities to the individual stiffnesses of springs linking information units to each other. Thereby an intermediate mapping of similarities onto parameters of a physically-based system is accomplished. Approaches of this type have already proved to provide promising results ([9], [10]). However, unlike any other method, we propose a *spherical embedding* of the similarity graph.

That is, given a set of objects $O_1, ..,O_n$ the central idea is to employ two concentric spheres for the initial positioning of information units as illustrated in Figure 1a. The objects are placed on the surface of the inner sphere and are attached with springs (anchors) to the *virtual* outer sphere and to each other. The major advantage of this arrangement is the degree of symmetry inherent

to the geometry and topology of a sphere. There are various problems to be solved upon initial object positioning and anchor stiffness assignment. First, a set of n equally spaced positions has to be computed on the surface of a sphere. This nontrivial optimization task, the so-called Fekete-problem [20], is approximated efficiently by a Poisson disc sampling in spherical coordinates. Then, objects are assigned to the samples using a breadth-first traversal strategy of the similarity graph. Note that the stiffness of the anchors has to balance the attractive forces of connected objects, where it controls the extent to which objects approach upon relaxation. The algorithmic details of this approach can be found in [16].
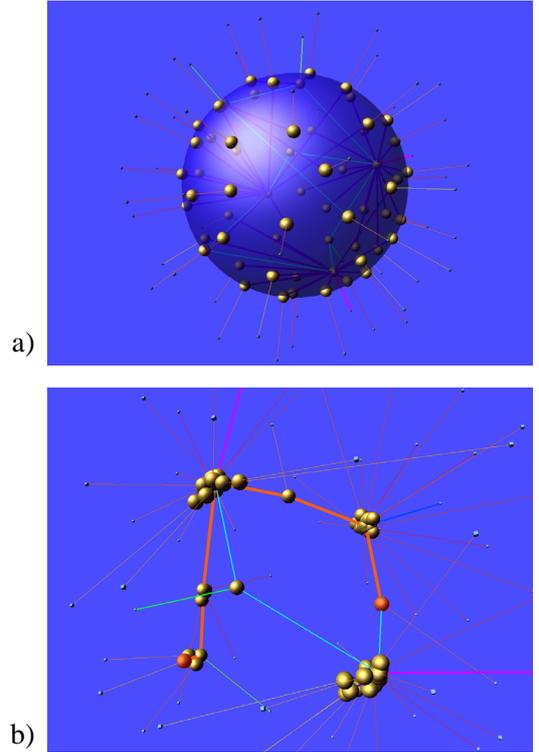

a)

b)

Figure 1: a) Initialization of objects by spherical embedding of the similarity graph. The outer sphere is not displayed. b) Object positions after relaxation of the mass spring system. The shortest path between two selected objects is highlighted.

For interactive analysis we exploit the graph topology of related documents in information space. This formalism enables us to carry over some of the fundamental algorithms [7] to support interactive analysis and clustering. For instance, we implemented minimal path procedures allowing the discovery and quantification of indirect links between objects.

4 HIERARCHY THROUGH CLUSTERING

Powerful clustering mechanisms to simplify the structure of complex subregions are essential to provide an efficient level-of-detail strategy.

Initial work for information visualization is reported in [6] who accomplished simple clustering by wrapping spheres around groups of objects. The transparency of the spheres was controlled as a function of the distance to the viewer. In [16] some of the authors of this paper proposed a K-means and PCA based clustering mechanism where the basic idea was to wrap ellipsoids around each object group whose shape is controlled by the principal components of the respective cluster. However, in both approaches the
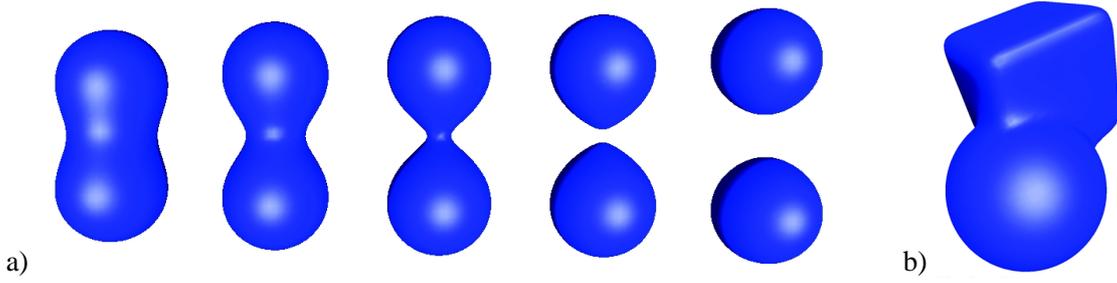
Figure 2: a) Illustration of implicit surfaces obtained from two field functions as their centroids are approaching. b) Modification of the shape by setting $\nu = \mu = 0.25$ for the upper field function.

quadric surface of the clustering hull represents an unnecessary restriction and may lead to inefficient use of space. Therefore, we developed a fast and simple clustering method which is described subsequently.

The fundamental idea is to use blob functions to represent individual objects and clustering shapes. Blobs, as initially been introduced to the graphics community by [17] allow to represent complex geometric shapes by implicit formulations. Here, a given volume is expanded by placing individual 3D basis functions $f_i(x,y,z)$ - so-called blobs - at some spatially scattered object locations (x_i, y_i, z_i). Introducing a set of weights b_i features an additional degree of freedom where the volume function $f(x,y,z)$ is obtained by the following linear expansion:

$$f(x, y, z) = \sum_{i=1}^{n} b_i \cdot f_i(x, y, z) \qquad (4)$$

An implicit definition of a 3D clustering isosurface is determined by selecting an isovalue c_{iso} with

$$f(x, y, z) = c_{iso} \qquad (5)$$

Note that for a physical interpretation the 3D basis function f_i can be considered describing some scalar field or potential. An appropriate choice of the bases is required for efficient handling of the blobs. In most cases, Gaussians or cubic splines provide good results. In a Gaussian representation of the field we have:

$$f_i(x, y, z) = e^{-a_i \cdot g_i(x, y, z)} \qquad (6)$$

where a_i stands for an individual scalar weight. For the exponent function g_i we adapted an approach proposed by [18] who essentially used a superquadric as

$$g_i(x, y, z) = ((x - x_i)^{2/\nu_i} + (y - y_i)^{2/\nu_i})^{\nu_i/\mu_i} + (z - z_i)^{2/\mu_i} \qquad (7)$$

where ν_i and μ_i are parameters to control the smoothness of the resulting shape. This formulation allows a maximum degree of freedom in modeling the implicit shape.
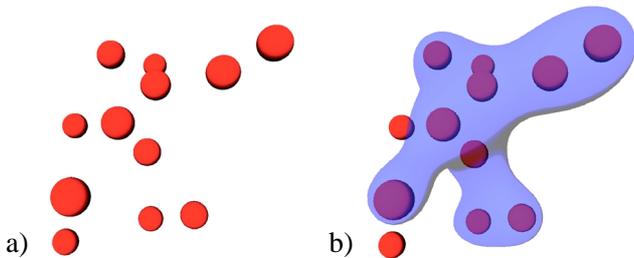

Figure 3: Clustering of a subset of objects scattered in 3D space: a) initial configuration, b) clustering isosurface as a transparent hull wrapped around the desired objects.

The effects of implicit clustering are illustrated by the picture series of Figure 2. Two initially disjoint isotropic (Gaussian) fields are clustered by an implicit isosurface which is computed by superimposing the associated basis functions. In the rightmost picture of the series the shape of the upper basis function was modified by tuning the superquadric parameters ν and μ, respectively. This property enables adapting individual clustering shapes to the underlying application scenario.

A more complex clustering problem is depicted in Figure 3 where a subset of scattered objects was wrapped by blobby clustering using the approaches described above. We observe that the implicit shape resulting from the individual field functions smoothly encapsulates all desired objects. Unlike PCA based clustering, almost no space is wasted.
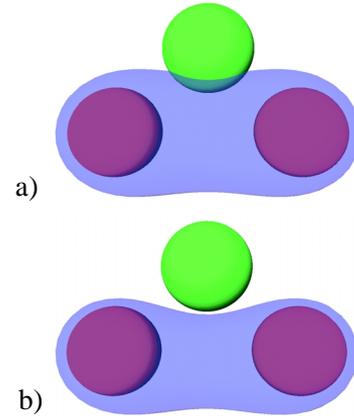

Figure 4: a) Undesired intersection of object and hull. b) repulse of the clustering surface from the object by assigning a negative value for b.

Yet, a problem arising with clustering is an undesired intersection of the hull with individual objects outside the cluster, such as presented in Figure 4a. Here, an interesting property of the blobby approach helps to solve the problem. By assigning a negative value to the weight b_i we generate an inhibiting field and the clustering hull is pushed away from the object (Figure 4b). Isosurfaces can be computed at interactive rates using appropriate techniques.

However, the blobby approach from above essentially performs a visual clustering. The automatic identification of subsets of objects has to be computed by additional procedures upon isosurface reconstruction.

We propose to modify an implicit surface polygonizer, such as the one presented in [21]. The following algorithm computes a semantic encapsulation of objects into sets of individual clusters under the following restrictions:

- *disjointness*: objects may belong to one and only one cluster. Multiple assignments are prohibited.
- *non-intersection of clusters*: clusters must not intersect. As illustrated in Figure 4, undesired intersections can be resolved using negative weights

The above preconditions guarantee a partitioning of objects into disjoint clusters encapsulated by disjoint isosurfaces. In this case the object belongs to the cluster whose surface is intersected first when traveling on a ray from the object center outwards. A pseudocode segment of the respective volume-based clustering algorithm can be given as follows:

```
Input(ciso)
For (all DataObjects O) do
    Search(first cell V intersected by cluster C, ciso)
    if(V already computed)
        // Object O belongs to cluster C
        add(O to children-list of C)
    else
        // Traverse recursively to compute isosurface
        traverse(surface C starting with V, ciso)
    fi
od
```

The algorithm starts from the center of a given object, and after discretization of the field function it searches the nearest volume cell intersecting the isosurface defined by c_{iso}. If the isosurface segments (triangles) of this voxel have already been computed, the surface segment exists and the object is added to the corresponding cluster. If not, the clustering hull (isosurface) is initialized and computed recursively with the cell as a seeding point.

Note that this method requires to store the visited cells. For efficient implementation, we recommend to build a hash table with keys computed from cell coordinates.

5 IMPLEMENTATION

The methods described in the previous sections are embedded in a prototype toolkit for information visualization, named IVORY, whose conceptual components are illustrated in the diagram of Figure 5.
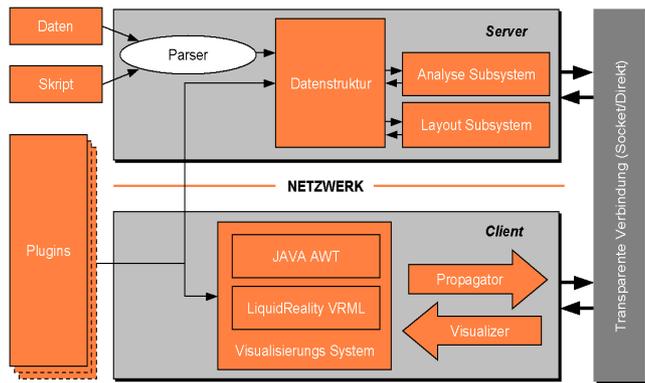

Figure 5:   Conceptual components of the information visualization toolkit IVORY.

The toolkit is divided into a backend server and a frontend client which are designed for distributed environments. The server essentially performs any computation related to the physically-based approach, such as solving the differential equations. Furthermore, various analysis methods, like shortest path search or clustering, are solved at the server side. A proprietary script language allows to include different data file formats. Remote object invocations accomplish a transparent client server connection. The cli-
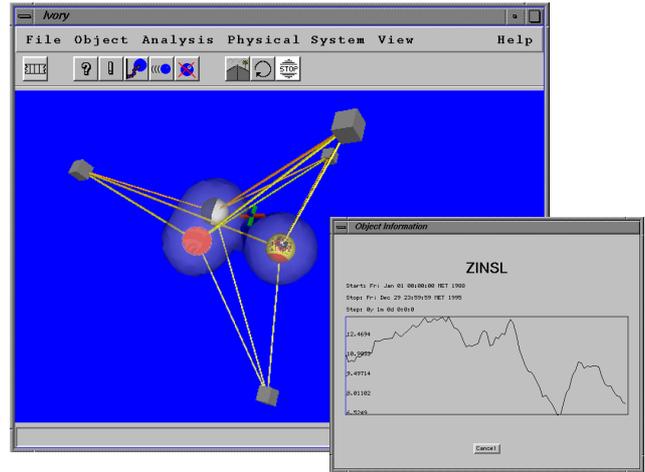

Figure 6:   Screenshot of the IVORY client under IRIX 6.2. Original data is displayed when picking the corresponding object.

ents' functionality comprises all issues related to the user interface and 3D rendering, where the JAVA AWT is used along with Dimension X's LiquidReality 3D rendering class library.

In our design we adopted the concept of plug-ins to configure the system for a particular application. The plug-in enables the user to define respective similarity metrics, the individual object appearance and behavior in VRML, interactive data editing, etc.

An additional VRML 2.0 export allows generation of scene files or animated sequences that can be inspected with almost every WWW browser [19].

IVORY is implemented in JAVA 1.0.2 and runs on both Linux/ Unix platforms and on Windows NT. A screenshot of IVORY is depicted in Figure 6. The presented plug-in enables, for instance, displaying the original data charts when picking a 3D object in the scene.
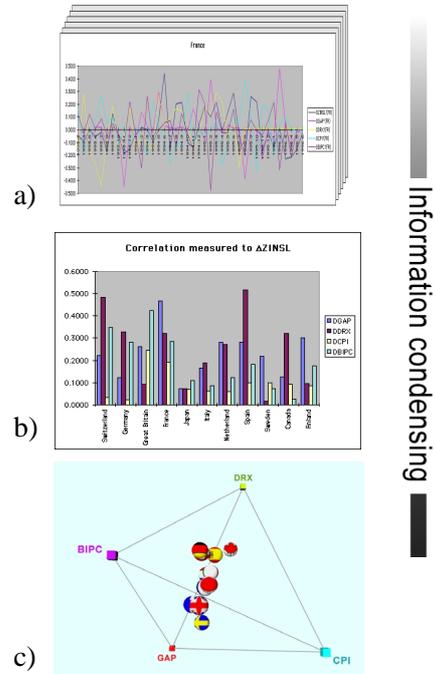

Figure 7:   Condensing multivariate relationships: a) stack of conventional diagrams b) correlation tables c) physically-based visualization paradigm.

6 EXAMPLE

The following section illustrates our approach by the visualization and analysis of long term interest rates correlating with other important economic parameters. Note, however, that the 2D pictures of this section do not reveal the full 3D arrangement computed by our method.

First, the method is compared to a traditional way of analyzing multidimensional relationships of economic indicators. The goal was to evaluate the influence of the indicators presented in the diagrams of Figure 7a on the long term interest rates of individual countries. Each of these indicators was computed relative to the **USA** as a reference. The state of the art approach, as depicted in Figure 7b, consists of producing bar charts showing the correlation with individual indicators for different countries. These charts form a basis for further interpretation performed by the financial analyst. In order to map the problem onto our visualization paradigm we start from a *special instance* of our model. By imposing displacement constraints we first generate a subset of objects which keep their position during relaxation. For visualization, our indicators are mapped on these object types. Conversely, we drop the anchor stiffnesses of all other objects to zero, that is we cut off their anchors. These freely movable objects represent the countries and are connected via links to all rigid objects from above. The spring stiffness of a link corresponds to the correlation of the associated indicator to the long term interest rates of this country. Note that the movable objects are not interconnected.
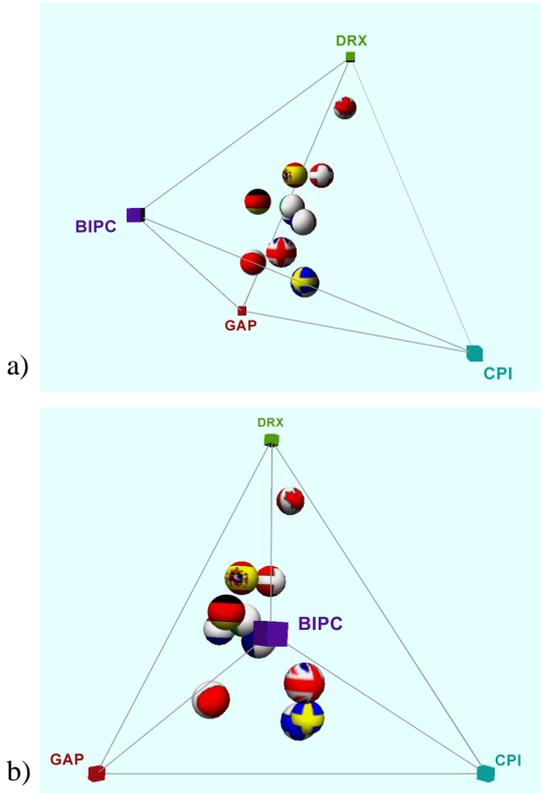

a)

b)

Figure 8: Two views showing the influence of individual economic indicators onto the long term interest rates of different countries (for VRML-files, readers might check out [19]; Data source: courtesy SBV, Basel, Switzerland)

Figure 8 displays two views of the relaxed model. The cubes at the vertices of the tetrahedral structure stand for the different indicators taken into account and the spheres representing the countries are textured with their flags. Although the definition of

individual indicators is beyond the scope of this paper we observe that the interest rates of Canada correlate tightly with the index DRX, whereas Switzerland relates more closely to GAP and CPI. Conversely, Germany is located near the center of gravity of the plane spanned by DRX, BIPC and GAP and is hence equally influenced by those measures.

The performance of the blobby clustering method is illustrated in the series of Figure 9, where different snapshots are presented as obtained from interactive settings of the isovalue. All blobs were equally weighted by a used in (6). Cluster sets of a single object are colored in red, those of two or three objects are in green whereas larger sets are rendered in blue. As expected, the system discriminates elegantly Canada - since heavily influenced by DRX - from the rest of the analyzed countries. Conversely, countries equally influenced by all indicators, such as France or Switzerland are grouped together in Figure 9b.
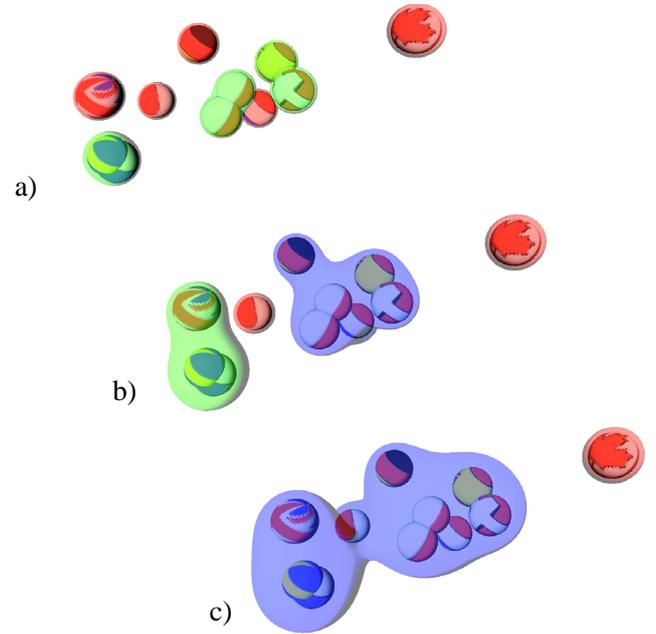

a)

b)

c)

Figure 9: Clustering the results of Figure 8: a) $a = 6.0$ and $c_{iso} = 0.1$, b) $a = 1.0$ and $c_{iso} = 0.9$, c) $a = 0.55$ and $c_{iso} = 1.1$.

CONCLUSIONS AND FUTURE WORK

We presented a new variant for physically-based information visualization. The fundamental idea is to arrange all information units on the inner part of two concentric spheres and to attach them with springs to each other. Relaxation of the model figures out the structural relations in information space. Additional clustering helps to simplify complex arrangements. Specifically, we are convinced that the physically-based approach fits nicely advanced I/O concepts with force and tactile feedback. Future research has to encompass a generalization of the physically-based approach including time-series and limited lifetime of particles.

ACKNOWLEDGMENT

REFERENCES

[1]   M. Gross, R. Koch: Visualization of Multidimensional Shape and Texture Features in Laser Range Data using Complex-Valued Gabor Wavelets, IEEE-Transactions on Visualization and Computer Graphics, Vol. 1, No. 1, pp.44-59, 1995

[2]   A. Witkin, D. Baraff, M. Kass: Physically-Based Modeling, SIGGRAPH Course Notes 34, 1995

[3]   J. Carriere, R. Katzman: Research Report - Interacting with Huge Hierarchies: Beyond Cone Trees, Proceedings of the IEEE Info. Vis. 95, pp. 74-81, 1995

[4]   J. Wise, et al.: Visualizing the Non-Visual: Spatial analysis and Interaction with Information from text Documents, Proceedings of the IEEE Info. Vis. 95, pp. 51-58, 1995

[5]   F. Young, P. Rheingans: Visualizing Structure in High-Dimensional Multivariate Data, IBM Journal of Research and Development, Vol. 35, No. 1/2, pp. 97-107, 1991

[6]   R. Hendley, et al.: Case Study - Narcissus: Visualizing Information, Proceedings of the IEEE Information Visualization 95, pp. 90-96, 1995

[7]   T. H. Cormen, C. E. Leiserson, and R. L. Rivest: Introduction to Algorithms. MIT Press, Cambridge, Massachusetts, 1994

[8]   I. Bruss, A. Frick: Fast Interactive 3-D Graph Visualization, Proceedings of Graph Drawing 95, Springer Verlag, LNCS 1027, p. 99-110, 1996

[9]   R. J. Hendley, N. S. Drew: Visualisation of complex systems, http://www.cs.bham.ac.uk/~rjh/aig.html, 1995

[10]   M. Chalmers: A Linear Time Layout Algorithm for Visualizing High-Dimensional Data, Proceedings of the IEEE Information Visualization 96, pp. 127-132, 1996

[11]   M. Harada, A. Witkin, D. Baraff: Interactive Physically-Based Manipulation of Discrete/Continuous Models, Proceedings SIGGRAPH 95, pp. 199-208, 1995

[12]   A. Wood, et al.: HyperSpace: Web Browsing with Visualisation. Third International World-Wide Web Conference, Poster Proceedings, Darmstadt, Germany, pp. 21-25, 1995

[13]   T. R. Henry, S. E. Hudson: Interactive Graph Layout. Proceedings ACM SIGGRAPH Symposium on UI Software, 1991

[14]   C. L. Bentley: Animating Multidimensional Scaling to Visualize N-Dimensional Data Sets, Proceedings of the IEEE Information Visualisation 96, pp. 72-73, 1996

[15]   S. Card, S. G. Eick, N. Gershon: Information Visualization, SIGGRAPH 96 Course Notes 8, 1996

[16]   T. C. Sprenger, M. H. Gross, A. Eggenberger, M. Kaufmann: A Framework for Physically-Based Information Visualization. Eight EuroGraphics-Workshop on Visualization in Scientific Computing, France, pp. 77-86, 1997

[17]   J. F. Blinn: A Generalization of Algebraic Surface Drawing. ACM Transactions on Graphics, pp. 235–256, July 1982

[18]   S. Muraki: Volumetric shape description of range data using "Blobby Model". Proceedings SIGGRAPH 91, pp. 227–235, July 1991

[19]   CGRG Homepage, ETH Zürich, http://www.inf.ethz.ch/department/IS/cg/html/research/infovis.html

[20]   János D. Pintér: Computation of Elliptic Fekete Point Sets. Internal report CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

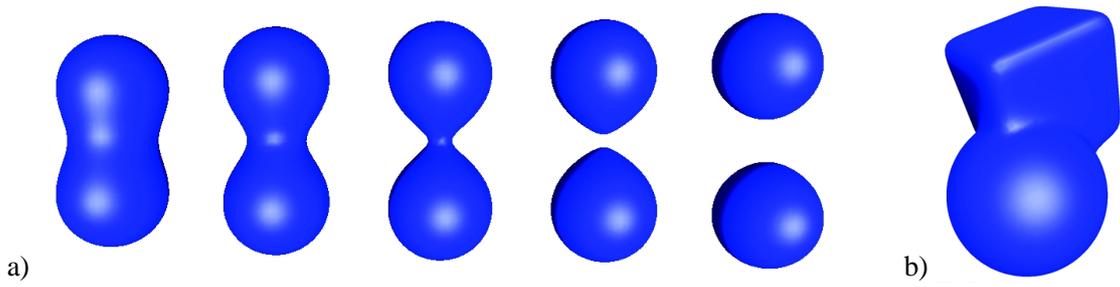[21]   J. Bloomenthal: An Implicit Surface Polygonizer. Graphics Gems IV Boston: Academic Press, pp. 324-349, 1994

Figure 2: a) Illustration of implicit surfaces obtained from two field functions as their centroids are approaching. b) Modification of the shape by setting $\nu = \mu = 0.25$ for the upper field function.
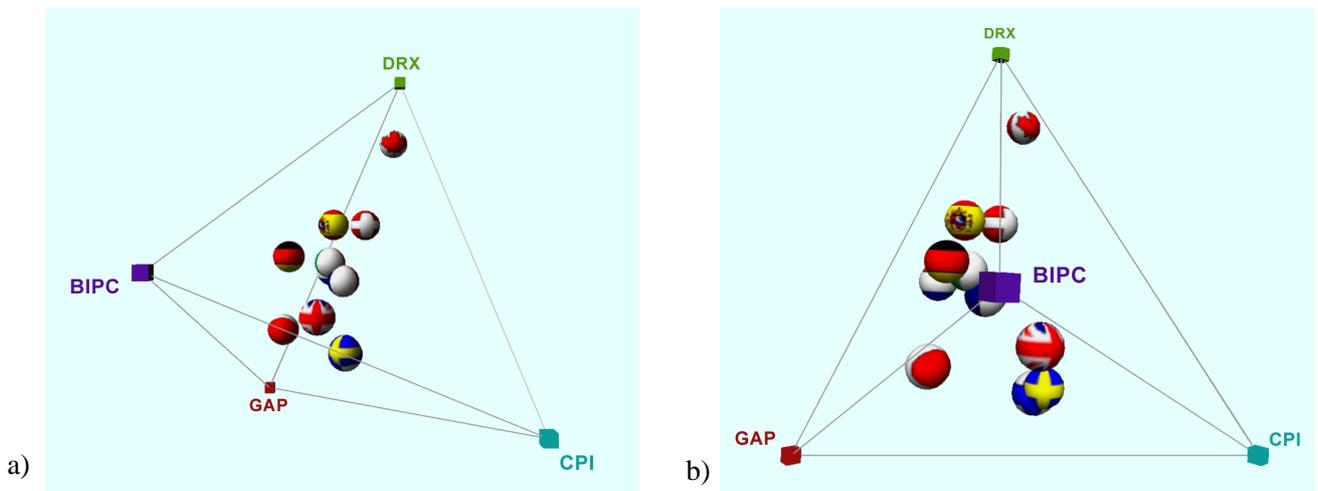

Figure 8: Two views showing the influence of individual economic indicators onto the long term interest rates of different countries (for VRML-files, readers might check out [19]; Data source: courtesy SBV, Basel, Switzerland)
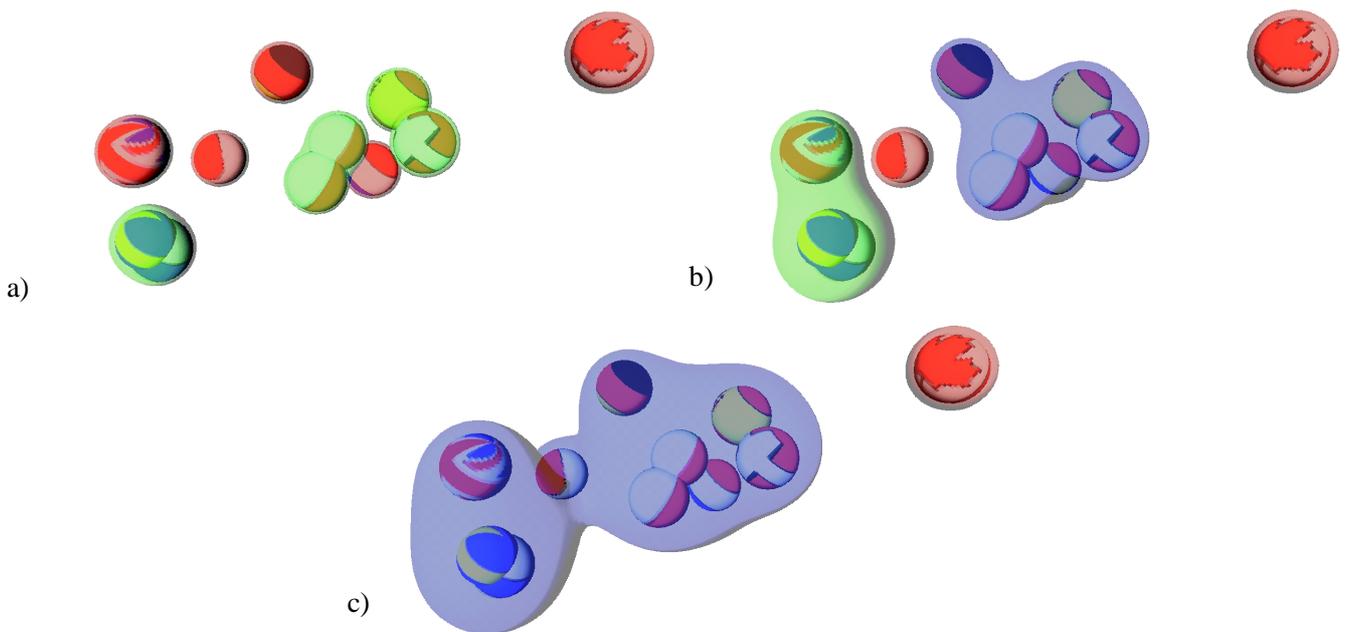

Figure 9: Clustering the results of Figure 8: a) $a = 6.0$ and $c_{iso} = 0.1$, b) $a = 1.0$ and $c_{iso} = 0.9$, c) $a = 0.55$ and $c_{iso} = 1.1$.