A Hybrid Approach for Multiresolution Modeling of Large-Scale Scientific Data

Tina Eliassi-Rad
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
P.O. Box 808, L-560, Livermore, CA 94551, USA
+1 925 422 1552
eliassi@llnl.gov

Terence Critchlow
Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
P.O. Box 808, L-560, Livermore, CA 94551, USA
+1 925 423 5682
critchlow@llnl.gov

ABSTRACT

Simulations of complex scientific phenomena involve the execution of massively parallel computer programs. These simulation programs generate large-scale multidimensional data sets over the spatio-temporal region. Analyzing such massive data sets is an essential step in helping scientists glean new information. To this end, efficient and effective data models are needed. In this paper, we present a hybrid approach for constructing data models from large-scale multidimensional scientific data sets. Our models not only provide descriptive information about the data but also allow users to subsequently examine the data by querying the data models. Our approach combines a *multiresolution-topological* model of the data with a *multivariate-physical* model of the data to generate one hierarchical data model that efficiently captures both the spatio-temporal and the physical aspects of the data. In particular, this hybrid approach consists of three phases. In the first phase, we build a multiresolution model that encapsulates the data set's spatial information (*i.e.*, topology and spatial connectivity). In the second phase, we build a multivariate model from the physical dimensions of the data set. Physical dimensions refer to those dimensions that are neither spatial (*x*, *y*, *z*) nor temporal (*time*). The exclusion of the spatial-temporal dimensions from the clustering phase is important since "similar" characteristics could be located (spatially) far from each other. Finally, in the third phase, we connect the multivariate-physical model to the multiresolution-topological model by utilizing ideas from information retrieval. The third phase is essential since the multivariate-physical model does not contain any topological information (without which the model does not have accurate spatial context information). Experimental evaluations on two large-scale multidimensional scientific data sets illustrate the value of our hybrid approach.

Categories and Subject Descriptors

G.3 [**Mathematics of Computing**]: Probability and Statistics – *Correlation and regression analysis, Multivariate statistics, Statistical computing.* H.2.8 [**Database Management**]: Database Applications – *Data mining, scientific databases.* H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Clustering, Selection process*. I.5.1 [**Pattern Recognition**]: Models: *Statistical*. I.5.3 [**Pattern Recognition**]: Clustering – *Similarity measures*.

General Terms

Algorithms, Management, Measurement, Performance, Experimentation.

Keywords

Multiresolution indices, topological models, multivariate clusters, information retrieval, large-scale scientific data sets.

1. INTRODUCTION

Utilization of massively parallel computer systems has enabled scientists to simulate complex phenomena. Examples of such complex phenomena are evolutions and explosions of stars (see Figure 1), aftermaths of earthquakes, *etc*. The computer programs that simulate these phenomena encode complex differential equations and produce tera-scale data sets over the spatio-temporal region. In order to glean information from such large-scale data sets, scientists need efficient and effective models for analyzing and examining data [2, 5, 9]. Analysis and examination of data include generating descriptive information, finding outliers, processing users' queries, *etc*. To this end, we have developed a hybrid approach for constructing multiresolution models from large-scale scientific simulation data.

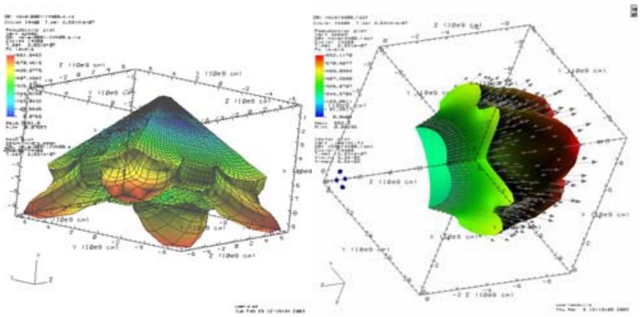

Figure 1. Snapshots of an Astrophysics Simulation Involving a Star's Explosion

SAC'05, March 13-17, 2005, Santa Fe, New Mexico, USA
Copyright 2005 ACM 1-58113-964-0/05/0003...$5.00

Effective models for scientific data sets must capture the temporal, spatial, and physical dimensions of the data. Physical dimensions[1] refer to those dimensions that are neither spatial (*x*, *y*, *z*) nor temporal (*time*) such as *temperature*, *pressure*, and *density*. Most existing models of multidimensional scientific data do not distinguish between spatial, temporal, and physical dimensions. Instead, they try to reduce the number of initial dimensions by using dimension reduction techniques (*e.g.*, principal component analysis) [13]. Our hybrid approach produces models that capture all dimensions of scientific data sets without dimension reduction. The central idea is to combine different models describing various characteristics of a large-scale multidimensional data set. In particular, we combine two models of the data. The first model is built solely from the spatial dimensions and the second model is constructed only from the physical dimensions. Our approach consists of three algorithms: (*i*) *spatial modeler*, which captures topology and spatial connectivity; (*ii*) *multivariate-physical modeler*, which captures the physical variables in multivariate clusters; and (*iii*) *physical-spatial linker*, which produces the hybrid data model. We trivially encode the discretized temporal dimension by constructing one data model per time step. Figure 2 provides a pictorial overview of our approach.

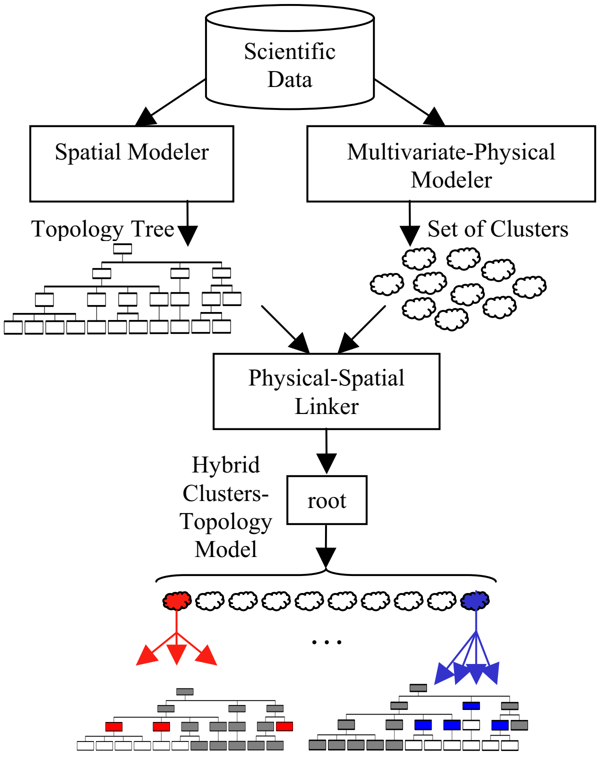

Figure 2. Pictorial Overview of Our Hybrid Approach

The spatial modeler constructs a multiresolution model that encapsulates a data set's spatial dimensions based on the intrinsic *topology*[2] of the data given in the original scientific problem. In particular, this modeler is a bottom-up algorithm, which iteratively agglomerates spatially connected regions [4].

To capture the dimensions of the data that are neither spatial nor temporal, we build multivariate clusters on *only* the physical dimensions of the data. The exclusion of the spatial dimensions from the clustering process is important since "similar" characteristics could be far from each other in the spatial region. Figure 3 illustrates this point with an example, in which the values of the physical variables in the outer layer of a star are homogeneous even though spatially the regions can be far apart.

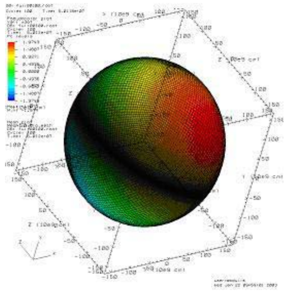

Figure 3. A Data Set Representing a Star (Similar Regions Have Similar Colors.)

To build multivariate clusters, we utilize a *smooth* clustering algorithm in conjunction with the *uncentered correlation coefficient* (*UCC*) as our similarity function [10]. Our choices for the clustering algorithm and the similarity function were influenced by (*i*) the sizes of our data sets (which consist of tens of millions of data points in multiple dimensions) and (*ii*) the importance of capturing both direction and magnitude of physical dimensions. The worst-case runtime of our clustering algorithm with UCC is O(k*n), where k is the number of resultant clusters and n is the total number of data points. In Section 4, we empirically show that on average k is several orders of magnitude smaller than n.

Since spatial dimensions do not play a role in building the multivariate clusters, it is important to associate each cluster with its precise spatial regions. In particular, it is essential that multivariate clusters frame answers to scientists' queries in the spatial location of the original data. To achieve this, the physical-spatial linker connects each multivariate cluster to the appropriate nodes of the multiresolution topological tree. We utilize ideas from information retrieval to establish such connections. The main challenge for physical-spatial linker is to find the best m nodes in the topology tree for each cluster, c, where m is much less than the number of data points in c.

This paper is organized as follows. Section 2 describes the data format for most scientific simulation data sets. Section 3 presents our approach for constructing hybrid multiresolution data models. In Section 4, we describe our experiments on two large-scale multidimensional scientific data sets. Sections 5 and 6 discuss some related and future works, respectively. Finally, Section 7 provides a summary of our work.

2. Scientific Simulation Data in Mesh Format

Most scientific simulation programs generate data in *mesh* format. Mesh data sets commonly contain *zones*, *time steps*, and *physical variables*. Zones are distinct spatial elements, which are

[1] We will use the terms *physical dimensions* and *physical variables* interchangeably.

[2] By topology, we mean the spatial connectivity of regions within the data set's discretized spatial dimensions.

generated from interconnected grids on the *x*, *y*, and *z* axes in the Euclidean space. The shapes of the zones can be regular (*e.g.*, rectilinear) or irregular (*e.g.*, arbitrary polygons). Each zone is identified by its *x*, *y*, and *z* coordinates. For examples, a cubic zone contains eight *x*, *y*, *z* triples, which identify its corners. Time steps are discrete steps in the temporal dimension. Since data changes over time, it is stored at different time steps. Physical variables denote non-spatio-temporal information. For each step in time, physical variables can be assigned values at a zone's corners or its center.

Figures 1 and 3 depict two mesh data sets produced by astrophysics simulations. Figure 1 represents an explosion of a star. Figure 3 depicts interactions of various components of a star at its mid-life. The three major factors determining the size of a mesh data set are its number of zones, time steps, and physical variables. Abdulla, *et al* [1] and Musick and Critchlow [14] provide nice introductions to scientific mesh data. Our approach is implemented for scientific data in mesh format but it can be used for modeling any multidimensional data set that contains data points/vectors on discretized spatio-temporal dimensions.

3. CONSTRUCTION OF HYBRID MULTIRESOLUTION MODELS FOR DATA ANALYSIS

This section describes the three components of our approach for constructing a hybrid multiresolution data model. They are (*i*) *spatial modeler*, (*ii*) *multivariate-physical constructor*, and (*iii*) *physical-spatial linker*.

3.1 Spatial Modeler

A precise spatial representation of a data set needs to capture the underlying topology of the original scientific problem. This topological information is stored in the connectivity of the data set's initial grid configuration (*i.e.*, at its zones). To this end, the immediate neighbors of each zone must be identified (see Figure 4a). Furthermore with large data, it is desirable to produce multiresolution models since they provide an efficient arrangement for the data [4]. To produce such models, our spatial modeler utilizes an iterative bottom-up agglomeration algorithm [9]. In particular, it employs a *coarsening* strategy that starts at a mesh data's initial grid configuration (see Figure 4b). From this fine level collection of grid cells, it iteratively produces *coarse* level collections of cells. The coarsening strategy performs a local heuristic search on the 2^N possible neighborhood configurations of a cell to find its connectivity (N = number of spatial dimensions).

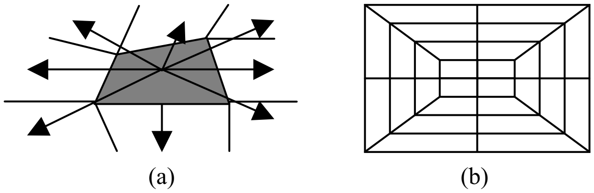

Figure 4. (a) The gray shape is an arbitrary zone in a 2D data set. The arrows point to the eight neighbors of the zone. (b) A rectilinear grid encoding a sphere (with edges glued together).

Figure 5 depicts the result of our coarsening strategy for a simple example. At the first iteration, the coarse cells (C1, C2, and C3 given by solid lines) are arranged from the twelve fine cells. At the next coarsening iteration, the cells C1 and C3 are agglomerated to produce cell C4. Cell C2 is not agglomerated in this iteration since it does not have a determinate right neighbor. At the last iteration, cells C2 and C4 are agglomerated to produce the root of the agglomeration tree.

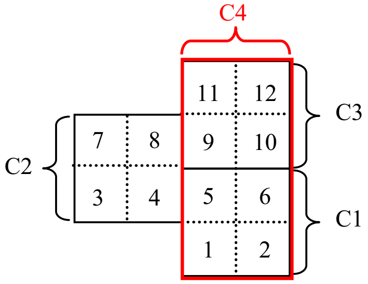

Figure 5. A non-quad tree coarse cell agglomeration

Since coarse cells are representations of fine-level collections of grid cells, the values of physical variables stored at fine-level cells are propagated into the coarse cells. In particular, for each physical variable within a coarse cell, we calculate its minimum, maximum, mean, and standard deviation from its values at the fine-level collection of cells [9]. These statistical values are then stored in the coarse cell.

3.2 Multivariate-Physical Modeler

Our multivariate-physical modeler constructs multivariate clusters. The motivation for creating multivariate clusters is to capture the interrelationships among a data set's physical variables in one metric. Such a metric enables us to collectively measure similarities between data points, from which we can provide high-level descriptive information about the data.

In multidimensional scientific data sets, each data point is (or can easily be converted to) a vector of values for physical variables defined over time and space [14]. That is, an (n+4)-dimensional data set consists of vectors of the following form: $(time, x, y, z, v_1, v_2, v_3, \cdots, v_n)$. In such cases, it is desirable to capture similarities in both direction and magnitude. For example, applications in physics usually produce data sets that contain measurements for velocity (see Figure 6). As such, if the goal is to find clusters of the data based on "similar" velocities, we will need to use a similarity function that encapsulates both direction and speed. To this end, we use the *uncentered correlation coefficient* (UCC) [8] with an approximately optimal offset [10], which is defined as follows:

$$UCC_{opt} = Sim\left(\vec{\alpha}, \vec{\beta}, \delta_{\vec{\alpha}}, \delta_{\vec{\beta}}\right) =$$

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\left(\vec{\alpha}_i - \delta_{\vec{\alpha}}\right)}{\sqrt{\frac{\sum_{j=1}^{n}\left(\vec{\alpha}_j - \delta_{\vec{\alpha}}\right)^2}{n}}} \times \frac{\left(\vec{\beta}_i - \delta_{\vec{\beta}}\right)}{\sqrt{\frac{\sum_{j=1}^{n}\left(\vec{\beta}_j - \delta_{\vec{\beta}}\right)^2}{n}}}\right), \text{ where}$$

$$\delta_{\vec{\alpha}} = \delta_{\vec{\beta}} = \frac{\mu_{\vec{\alpha}} + \mu_{\vec{\beta}}}{2}.$$

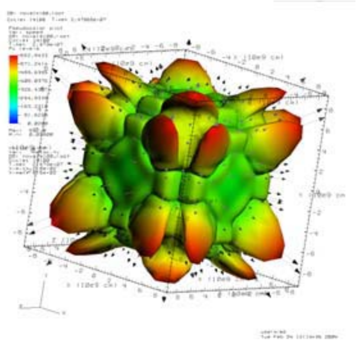

Figure 6. Velocity of a Star Exploding

Using UCC_{opt} eliminates the need for computing two metrics to measure similarities in direction (*i.e.*, angles between vectors) and similarities in magnitude (*i.e.*, distances between vectors). This is important especially when dealing with large-scale data sets.

We use a *smooth* (*a.k.a.*, *canopy*) clustering algorithm to create multivariate physical clusters (see Table 1) [15]. Our smooth clustering algorithm does not require the total number of clusters, k, to be given *a priori*. Instead, the given similarity threshold restricts our clusters. By fixing such a threshold, we are able to isolate the quality of our resultant clusters solely based on offsets given for the uncentered correlation coefficient.

Our clustering criterion is a conjunction of two conditions: (1) the similarity based on UCC_{opt} must be meet, and (2) the cluster's cohesion[3] with a newly added data point must be at most the same without the new addition. Our experimental results show that the second criterion reduces the running time of the clustering algorithm by a half. The reduction in running time is attributed to our placement procedure. Specifically, among clusters that can accept a given data point, we choose the first one that is "good enough" based on the cohesion constraint (as opposed to finding the optimal cluster to place the new data point). In addition the differences between clusters produced with and without the second criterion are not statistically significant (see Section 4).

The worst-case runtime of our clustering algorithm with UCC_{opt} is O(k*n), where k is the number of resultant clusters and n is the total number of data points. In Section 4, we empirically show that on average k is several orders of magnitude smaller than n.

Table 1. Multivariate-Physical Modeler

```
Inputs:
  • A collection of vectors, Vectors, representing the physical
    variables (i.e., dimensions) of zones in the data
  • A clustering (i.e., similarity) threshold, SimThreshold
  • An offset for the uncentered correlation coefficient, offset

Output:
  • A collection of clusters, Clusters

Algorithm:
  a) Clusters ← {};
  b) MaxCohesion ← −∞;
  c) For each vector, $\vec{\alpha}$, in Vectors do
     i) If (Clusters is empty)
        (1) Create a new cluster, C;
        (2) Add $\vec{\alpha}$ to C;
        (3) Create C's center, $\vec{\varsigma}$;
        (4) Add C to Clusters;
     ii) Otherwise
        (1) For each cluster, $C_i$, in Clusters
            (a) CurrCohesion = Cohesion($C_i$)
            (b) If (CurrCohesion > MaxCohesion)
                (i) MaxCohesion ← CurrCohesion;
            (c) NewCohesion ← Cohesion($C_i$ with $\vec{\alpha}$)
            (d) If ( $Sim(\vec{\alpha}, \vec{\varsigma}, offset, offset)$ ≥SimThreshold)
                and (NewCohesion ≤ MaxCohesion)
                (i) Add $\vec{\alpha}$ to C;
                (ii) Update $C_i$'s center, $\vec{\varsigma}$
                (iii) Break;
            (e) Otherwise
                (i) Create a new cluster, C;
                (ii) Add $\vec{\alpha}$ to C;
                (iii) Create C's center, $\vec{\varsigma}$;
                (iv) Add C to Clusters;
```

3.3 Physical-Spatial Linker

Even though spatial variables do not play a role in building our multivariate-physical clusters, it is desirable to associate each cluster with its precise spatial region. In particular, it is important for the clusters to return answers to scientists' queries in the spatial region of the original mesh. Since mesh data typically have millions of zones, it would be very inefficient (and at times impossible) to link each cluster with all of its zones. Therefore, we present a linking algorithm for connecting each cluster to a small set of nodes in the topology tree. Recall that a data set's topology tree is a multiresolution model that stores the spatial information of a data set by utilizing the intrinsic topology of the data given in the original scientific problem (see Section 3.1). To accomplish this task, we utilize ideas from the field of information retrieval. In particular, given a cluster $C=\{c_1, c_2, c_3, \ldots, c_m\}$ and a topology node $T = \{t_1, t_2, t_3, \ldots, t_n\}$, we define *precision* and *recall* as follows (see Figure 7):

$$precision(C, T) = \frac{|C \cap T|}{|T|} \text{ and } recall(C, T) = \frac{|C \cap T|}{|C|}$$

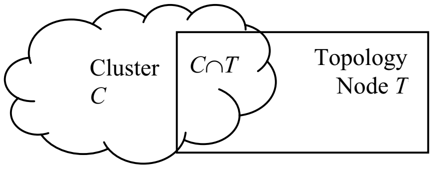

Figure 7. Clusters and Topology nodes share zones.

Table 2 describes the physical-spatial linker. A link is established between a cluster and a topology tree node when at least one the following conditions are true: (*i*) the *breakeven point* is satisfied (*i.e.*, precision is equal to recall), or (*ii*) precision is greater than a user-defined precision threshold. Obviously, links with both high precision and recall values are desired. The test for break-even point insures that such links are selected even if their precision does not satisfy the user-specified threshold. Such "good quality" links act as shortcuts into potentially tall topology trees.

[3] See Section 4.2 for a formal definition of cluster cohesion.

Table 2. Physical-Spatial Linker

```
Inputs:
    • A collection of zones, $Z$
    • A topology tree, $T$
    • A precision threshold, $PrecThreshold$
    • An offset for the uncentered correlation coefficient, $offset$
Output:
    • A collection of clusters, $Clusters$, and links connecting $Clusters$ to the topology tree, $T$
Algorithm:
  1) For each zone, $z_i$, in $Z$ do
     a) Place $z_i$ in the appropriate cluster, $C_j$ (see Table 1);
     b) Update the count for occurrences of $C_j$ in $z_i$'s ancestors within the topology tree, $T$;
  2) Iterate through the unexamined nodes, $T_i$, of the topology tree in a depth-first order
     a) For each cluster, $C_j$, do
        i) If ( ($precision(C_j, T_i) \geq recall(C_j, T_i)$) or
              ($precision(C_j, T_i) \geq PrecThreshold$) )
           (1) Establish a link between $C_j$ and $T_i$;
           (2) Store $precision(C_j, T_i)$ with the established link;
           (3) Mark the subtree rooted by $T_i$ as examined for $C_j$
               (i.e., do not iterate through the subtree rooted by $T_i$ for $C_j$);
```

4. EXPERIMENTS

This section describes the performance of the multivariate-physical modeler and the physical-spatial linker. Experiments for spatial-modeler can be found in [4].

4.1 Data Sets

Table 3 describes the two large mesh data sets used in our experiments. Recall that mesh data sets vary with time, consist of multiple dimensions (*i.e.*, variables), and contain interconnected spatial grids. Such grids break the mesh data into *zones*, in which data points are stored. The shape of the zones can be regular (*e.g.*, rectilinear) or irregular (*e.g.*, arbitrary polygons).

Table 3. Characteristics of Our Data Sets

Data Sets	# of Zones per Time Step	# of Variables per Zone	# of Time Steps
White Dwarf	557,375	29	22
Djehuty-5	1,625,000	27	16

Both mesh data sets are astrophysics simulations of a star at a certain stage of its life and represent readings in point locations of a continuous medium. The data sets are represented as zones. In these data sets, zones are small cubes with 8 nodes. Values of variables are associated either with each node of a zone (called a *nodal variable*) or with the center of each zone (called a *zonal variable*). The White Dwarf data set (see Figure 1) is a simulation of a star exploding. The Djehuty data set (see Figure 3) is a simulation of a star at its mid-life.

4.2 Results on Multivariate-Physical Modeler

Tables 4 and 5 list the results of the multivariate-physical clusters for White Dwarf and Djehuty-5, respectively. All values for physical variables are normalized to fall within 0 and 1. The following performance metrics are used in these tables:

- *Intracluster cohesion* measures the compactness of the clusters by utilizing the trace of the covariance matrix.[4] In other words, we sum the variances of each cluster across all of its dimensions [12]. The following equation provides a formal definition for our intracluster cohesion metric for any cluster C with a center $\vec{\varsigma}$: $cohesion(C) = \sum_{i=1}^{n} Variance(\vec{\varsigma}, i)$, where n is the number of physical dimensions in the data set. Compact clusters have small intracluster cohesion numbers. The average cluster cohesion corresponds to the average value of cluster cohesions for a particular time step.

- Intercluster separation is between cluster variations. We measure cluster separation by utilizing the weighted covariance of cluster means. In particular, we sum the weighted squared differences between cluster means [12]. The following equation provides a formal definition for our intercluster separation metric:

$$separation(C) = \sum_{k=1}^{c} \left(\frac{|C_k|}{\sum_{i=1}^{c} |C_i|} (\mu_k - \bar{\mu})(\mu_k - \bar{\mu})^T \right)$$

  , where $|C_k|$ is the number of vectors in cluster C_k, $\sum_{i=1}^{c} |C_i|$ is the total number of vectors in the data set, and $(\mu_k - \bar{\mu})$ is a vector of differences between cluster C_k's mean vector and the global mean vector of all the data points. Well-separated clusters have large intercluster separation numbers. The average cluster separation corresponds to the average value of cluster separations for a particular time step.

- Cluster quality measures the general quality of a cluster by the ratio of its intercluster separation to intracluster cohesion (*i.e.*, $\frac{separation}{cohesion}$). Higher values for the overall quality indicate better performance. The normalized average cluster quality corresponds to the average value of cluster qualities for a particular time step normalized by its number clusters.

Table 4. White Dwarf's Multivariate Clusters with Similarity Threshold of 0.95 (where 1 Denotes Complete Similarity).

Time Step	# of Clusters	Avg Cluster Cohesion	Avg Cluster Separation	Normalized Avg Cluster Quality
0	31	**0.0031**	3.75	**38.48**
1	48	0.0040	3.72	19.25
2	30	0.0173	3.72	7.19
3	31	0.0250	3.79	4.89
4	32	0.0251	3.73	4.65

[4] The trace of the covariance matrix is the sum of its diagonal entries. The diagonal entry in a covariance matrix, $Cov(X, X)$, is equivalent to the variance of X.

5	33	0.0201	3.75	5.65
6	40	0.0182	3.81	5.25
7	44	0.0120	3.72	7.03
8	47	0.0113	3.75	7.04
9	44	0.0087	3.80	9.94
10	48	0.0077	3.66	9.95
11	39	0.0195	3.84	5.06
12	43	0.0182	3.86	4.93
13	39	0.0218	4.02	4.74
14	41	0.0199	3.55	4.35
15	41	0.0209	3.58	4.19
16	41	0.0203	3.55	4.26
17	41	0.0206	3.55	4.20
18	38	0.0285	3.77	3.48
19	40	0.0215	**4.08**	4.75
20	37	0.0214	3.93	4.98
21	38	0.0199	3.87	5.11

Table 5. Djehuty-5's Multivariate Clusters with Similarity Threshold of 0.95 (where 1 Denotes Complete Similarity).

Time Step	# of Clusters	Avg Cluster Cohesion	Avg Cluster Separation	Normalized Avg Cluster Quality
0	247	0.0036	**3.69**	4.11
1	66	0.0075	3.54	7.12
2	79	**0.0017**	3.12	**23.72**
3	83	0.0046	3.28	8.52
4	80	0.0026	3.29	15.80
5	85	0.0040	3.15	9.32
6	90	0.0026	3.26	13.79
7	96	0.0016	3.28	21.52
8	94	0.0021	3.21	16.56
9	70	0.0028	3.38	17.13
10	68	0.0040	3.34	12.24
11	90	0.0031	3.10	11.11
12	89	0.0023	3.13	15.44
13	87	0.0027	3.16	13.69
14	96	0.0024	3.23	13.81
15	56	0.0052	3.52	12.14

Variations in the number of clusters across time steps represent changes in a star as it explodes (Table 4) or evolves (Table 5). In Tables 4 and 5, the values for the best average cluster cohesion, separation, and quality are boldfaced. In both data sets, clusters with the best average cluster cohesion also have the best average cluster quality. However, clusters with the best average separation do not have the best average cluster quality. This outcome is not unexpected since our multivariate-physical modeler is biased toward cluster cohesion. Due to limitations in space, we do not list the results without the maximum cohesion criteria for our data sets. However, on our data sets, the two-sided tests with normal and student *t*-distributions on mean differences between results with and without the maximum cohesion criterion are not statistically significant at 95% for (*i*) number of clusters and (*ii*) normalized average cluster quality. Recall that we utilize this cohesion criterion in order to speed-up the clustering algorithm by not searching for the optimal cluster to put a new data point. Instead a new data point is placed in the first cluster that (*i*) meets the user's similarity threshold and (*ii*) its cohesion does not increase with the new addition. In our experiments, the runtime was reduced by half (which is usually several hours).

4.3 Results on the Physical-Spatial Linker

This section depicts the results of our physical-spatial linker. It shows how effectively multivariate-physical clusters can be connected to topology tree nodes to create one hybrid data model.

The topology trees constructed for White Dwarf and Djehuty-5 have 10 levels (*i.e.*, the height from root to leaves). As mentioned in Section 3.1, minimum, maximum, mean, and standard deviation values are stored at each node in the topology tree. Model error in these nodes is a function of standard deviation [4]. As one moves from the leaves toward the root of a topology tree, model errors in the nodes increase because standard deviation values increase.

4.3.1 Ratio of Links to Cluster Items

Figures 8 and 9 illustrate average ratio of the number of links between clusters and topology nodes to the number of items in clusters for White Dwarf and Djehuty-5, respectively. As the precision threshold increases, more links are required and so the ratio gets closer to 1. However, note the small increase in the ratio as the precision threshold is increased from 0.25 to 0.5 as opposed to the more substantial increase in the ratio as precision threshold is increased from 0.5 to 0.75. This outcome is expected since as precision gets closer to one, links can only be established between cluster items and topology tree nodes that represent the same underlying data. That is, there does not exist a linear relationship between precision and the number of physical-spatial links.

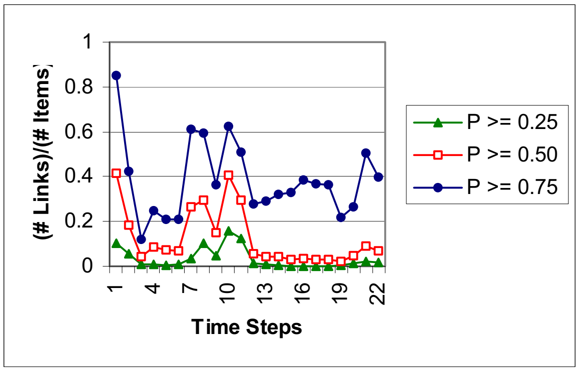

Figure 8. White Dwarf: Ratio of Physical-Spatial Links to Cluster Items for Various Precision Thresholds

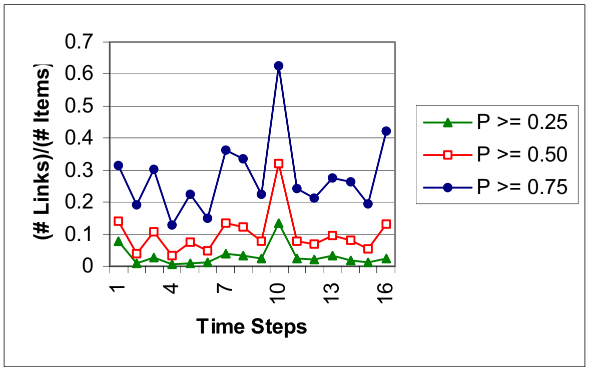

Figure 9. Djehuty-5: Ratio of Physical-Spatial Links to Cluster Items for Various Precision Thresholds

4.3.2 Number of Physical-Spatial Links

Figures 10 and 11 depict the maximum number of links between clusters and topology nodes for various precision thresholds on White Dwarf and Djehuty-5, respectively. They support the results and discussion in Section 4.3.1. As precision approaches one, the number of required links for connecting the multivariate-physical model to the spatial model increases nonlinearly because each link is forced to include less cluster items.

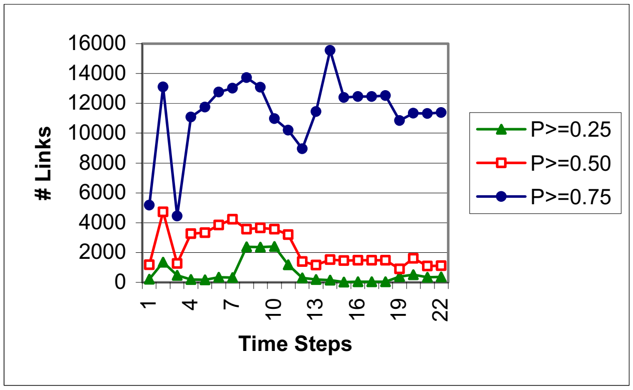

Figure 10. White Dwarf: Maximum Links between Clusters and Topology Nodes for Various Precision Thresholds

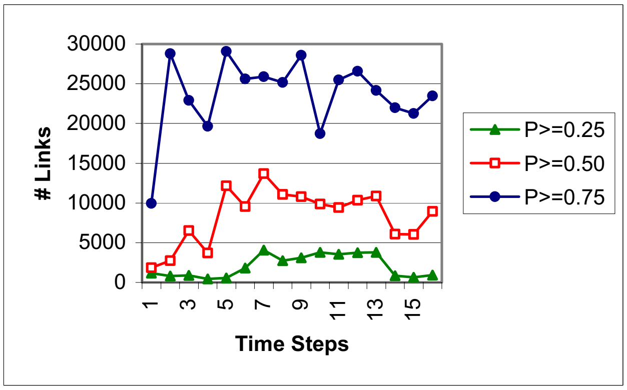

Figure 11. Djehuty-5: Maximum Links between Clusters and Topology Nodes for Various Precision Thresholds

4.3.3 Levels into Topology Tree

Figures 12 and 13 show the normalized levels of the topology tree accessed by physical-spatial links for various precision thresholds on White Dwarf and Djehuty-5, respectively. Values close to 1 indicate levels near the leaves of a tree. Values close to 0 denote levels near a tree's root. Recall that the trees each have 10 levels.

Links that are high in precision and point to levels close to the tree's root are desirable. As these figures depict, there is an intrinsic tradeoff between links having high precision and pointing to levels near the root. In other words, we see the usual tradeoff between precision and recall. Finally, it is interesting to note that the nonlinear relationships described in Section 4.3.1 (between precision and the ratio of links to cluster items as precision nears one) and Section 4.3.2 (between precision and number of links as precision nears one) are absent here. In this case, the nonlinear relationships are visible as precision approaches zero. A precision threshold of 0.25 is not close enough to zero to show these nonlinear relationships.

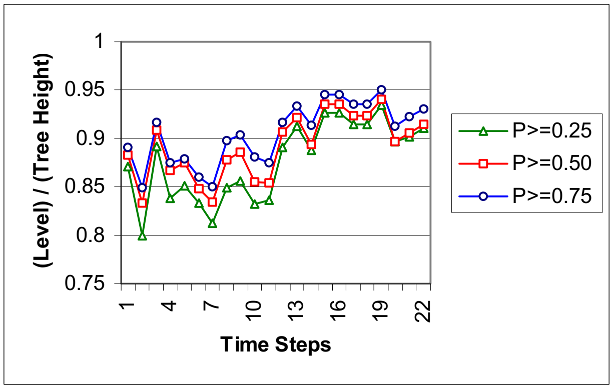

Figure 12. White Dwarf: Ratio of Topology Tree Levels Accessed for Various Precision Thresholds

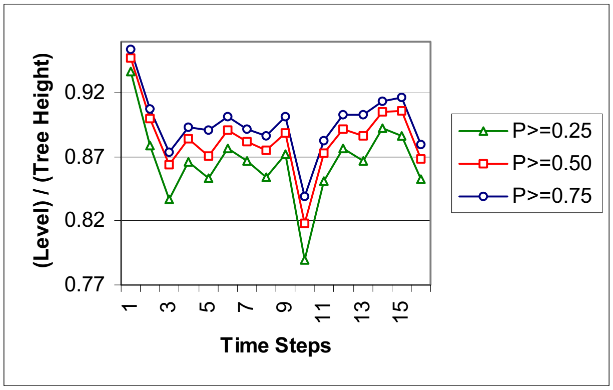

Figure 13. Djehuty-5: Ratio of Topology Tree Levels Accessed for Various Precision Thresholds

5. RELATED WORK

Little research has been done on similarity metrics that capture both direction and magnitude. Uncentered correlation coefficients are mostly popular in microarray analysis [6, 8].

Popular clustering algorithms [16] (such as DBSCAN) cannot be used or scaled to large-scale scientific simulation data sets for one or more of the following reasons:

1. Our modeling techniques cannot require sampling. Scientists already sample the data produced by their simulation programs. They do not accept models that sample from the sampled data, particularly since they are mostly interested in outliers (which are either bugs in their programs or new scientific discoveries).

2. We cannot build clusters from zones in a subspace of the data since global properties are important.

3. It is not desirable to use binning or histograms techniques since we are not supposed to assume an *a priori* distribution on the data. Moreover, histograms are computationally expensive on high-dimensional data sets.

Our work is similar to Freitag and Loy [11]. Their system builds distributed octrees from large scientific data sets. However, they reduce their data by constraining the points to their spatial locations. This strategy does not allow for grouping of data points with similar physical values that are spatially far from each other.

STING [17] is also similar to our work except that it assumes that the distribution of the data is known. Also, it has been tested only on small data sets containing only tens of thousands of data points. DuMouchel, *et al* [7] present a method for compressing flat files; however, they use binning techniques to "squash" files, which impose *a priori* distributions on the data Finally, AQUA [3] uses cached summary data in an OLAP domain. They also use sampling and histogram techniques, which are not acceptable in our models.

6. FUTURE WORK

We are investigating other modeling techniques for large-scale simulation data sets. Specifically, we are interested in models that (*i*) require only one sweep of data, (*ii*) are good at finding outliers, (*iii*) can be easily parallelized, and (*iv*) can efficiently answer a wide variety of queries. In addition, we are examining other criteria for encapsulating direction and magnitude. We intentionally did not discuss the simple criteria of using a weighted combination of popular similarity and dissimilarity metrics since appropriate selection of weights for the two metrics can be tricky. Finally, we plan to develop tools, which track a particular zone across time steps. Such tools will not only help scientists' in their investigation but also will provide us with insights into our modeling algorithms.

7. CONCLUSION

Massively parallel computer programs (which simulate complex scientific phenomena) generate large-scale data sets over the spatio-temporal region. Analyzing such massive data sets is an essential step in helping scientists glean new information. To this end, efficient and effective data models are needed. In this paper, we presented a hybrid approach for constructing data models from large-scale multidimensional scientific data sets. Our data models not only provide descriptive information about the data but also allow users to examine the data further by querying the data models. Our approach combines a multiresolution-topological model with a multivariate-physical model to generate one hierarchical data model that efficiently captures both the spatio-temporal and the physical aspects of the data. The exclusion of the spatial-temporal dimensions from the multivariate-physical models is important since "similar" characteristics could be located (spatially) far from each other. We connect the multivariate-physical model to the multiresolution-topological model by utilizing ideas from information retrieval. Experimental evaluations on two large-scale multidimensional astrophysics data sets illustrate the value of our hybrid data model in capturing the evolution and explosion of a star through the combination of multivariate-physical clusters and a topology tree. Finally, our approach confirms the notion that a combination of multiple models, which describe different characteristics of a data set, is effective on large-scale multidimensional scientific data.

8. ACKNOWLEDGMENTS

This work was performed under the auspices of the U.S. Department of Energy by the University of California Lawrence Livermore National Laboratory under contract No. W-7405-ENG-48.1. UCRL-CONF-206419. Our thanks to Ghaleb Abdulla, Bill Arrighi, Chuck Baldwin, Susan Hazlett, and Megan Thomas for their assistance.

9. REFERENCES

[1] Abdulla, G., Critchlow, T., Arrighi, W. Simulation Data as Data Streams, In *SIGMOD Record*, *33*, 1 (March 2004).

[2] Abdulla, G., Baldwin, C., Critchlow, T., Kamimura, R., Lozares, I., Musick, R., Tang, N.A., Lee, B., and Snapp, R. Approximate ad-hoc query engine for simulation data, In *JCDL 2001,* 255-256.

[3] Acharya, S., Gibbsons, P.B., Poosala, V., and Ramaswamy, S. The Aqua approximate query answering system, In *ACM SIGMOD 1999*, 574-576.

[4] Baldwin, C., Eliassi-Rad, T., Abdulla, G., and Critchlow, T. The evolution of a hierarchical partitioning algorithm for large-scale scientific data: three steps of increasing complexity, In *SSDBM 2003*, 225-228.

[5] Baldwin, C., Abdulla, G., Critchlow, T. Multi-resolution modeling of large scale scientific simulation data, In *CIKM 2003*, 40-48.

[6] Dadgostar, H., Zarnegar, B., Hoffmann, A., Qin, X.-F., Truong, U., Rao, G., Baltimore, D., and Cheng, G., Cooperation of multiple signaling pathways in CD40-regulated gene expression in B lymphocytes. In *Proc. of National Academy of Sciences of the U.S.A., 99*, 3, 2002, 1497-1502.

[7] DuMouchel, W., Volinsky, CH., Johnson, T., Cortes, C., and Pregibon, D., Squashing flat files flatter, In *KDD 1999*, 6-15.

[8] Eisen, M.B., Spellman, P.T., Brown, P.O., and Botstein, D. Cluster analysis and display of genome-wide expression patterns. In *Proc. of the National Academy of Sciences of the U.S.A., 95*, 25, 1998, 14863-14868.

[9] Eliassi-Rad, T., Baldwin, C., Abdulla, G., and Critchlow, T. Statistical modeling of large-scale scientific simulation data. *New Generation of Data Mining Applications*, Eds: Zurada J. and Kantardzic M., IEEE Press/Wiley, January 2005.

[10] Eliassi-Rad, T., and Critchlow, T. Clustering with Uncentered Correlation Coefficients: Beware of Offsets, Lawrence Livermore Technical Report, 2004.

[11] Freitag, L.A., and Loy, R.M. Adaptive, multi-resolution visualization of large data sets using a distributed memory octree, *Supercomputing 1999*, Article 60.

[12] Hand, D., Mannila, H., and Smyth, P. *Principles of Data Mining*, MIT Press, Cambridge, MA, 2001.

[13] Jolliffe, I. T. *Principal Component Analysis*, Springer-Verlag; 2nd edition, 2002.

[14] Musick, R., and Critchlow, T. Practical lessons in supporting large-scale computational science, In *SIGMOD Record*, *28*, 4 (December 1999).

[15] Ng, R.T., and Han, J., Efficient and effective clustering methods for spatial data mining, In *VLDB 1994*, 144-155.

[16] Parsons, L., Haque, E., and Liu, H. Subspace Clustering for High Dimensional Data: A Review. In *SIGKDD Explorations, 6,* 1 (June 2004), 90-105.

[17] Wang, W., Yang, J., and Muntz, R. STING: A statistical information grid approach to spatial data mining, In *VLDB 1997*, 186-195.