Scenario Analysis and PRA: Overview and Lessons Learned

Diego Mandelli, Tunc Aldemir
Nuclear Engineering Department
The Ohio State University, USA

Alper Yilmaz
Photogrammetric Computer Vision Laboratory
The Ohio State University, USA

ABSTRACT: The recent trend to use a best estimate plus uncertainty (BEPU) approach to nuclear reactor safety analysis instead of the traditional conservative approach can lead to the production of very large amounts of data. Therefore the need for methodologies that can facilitate the analysis of high volumes of data in terms of both the cardinality (due to the high number of uncertainties included in the analysis) and the dimensionality (due to the complexity of systems) arises. Clustering methodologies offer powerful tools that can help the user to identify scenario groups that are representative of the data and thus can reduce the effort involved in data analysis. In this paper, we consider data that are generated using the dynamic event tree approach for nuclear power reactor transients which contain a large set of state variables (e.g., temperature, pressure of specific nodes in the simulator) and information regarding the status of specific components/systems. Techniques for clustering of the raw data, using in particular Mean-Shift methodology, are discussed and evaluated. We highlight the lessons learned from the research activities at The Ohio State University and possible future research directions. In addition, pre-processing of the raw data and dimensionality reduction techniques are described and compared using several examples.

1 INTRODUCTION

The new generation of safety analysis codes[1] will incorporate a new series of algorithms and dynamic probabilistic safety assessment methodologies (Aldemir, Catalyurek, Denning, Smidts, Sun, & Yilmaz 2011) that are able to

- model system dynamics,
- model human interaction and digital control systems, and,
- perform uncertainty quantification and sensitivity analysis.

An objective in this trend is to use a best estimate plus uncertainty (BEPU) approach to nuclear reactor safety analysis instead of the traditional conservative approach. This process may result in large amounts of data generated that can be difficult to analyze and, from a user point of view, might be difficult to assess with regard to the main contributors to risk and most relevant trends.

Clustering methodologies (Bishop 2007) offer powerful tools that can help the user to identify scenario groups that are representative of the data. In the nuclear industry, the data analysis problem has been tackled by using *classification* algorithms (Zio & Baraldi 2005, Mercurio, Podofillini, Zio, & Dang 2009). Clustering differs from classification from the fact that it is an unsupervised type of classification where classes are not predefined. Classification, on the other hand, is a supervised methodology where classes are defined *a priori* by the user.

This article will summarize the research activities carried out at The Ohio State University (OSU) in the past few years towards the development of clustering algorithms that can reduce the complexity of data analysis efforts. In Sections 2 and 3 we will introduce the clustering problem and give an overview of the major steps that

[1] LWR Sustainability Program (INL/EXT-07-13543, "Strategic Plan for Light Water Reactor Research and Development," Idaho National Laboratory, November 2007).

are required to analyze data correctly. Section 4 provides examples of some methodologies available in order to pre-process the raw data before the clustering step. This section also shows how each scenario can be represented and how dimensionality can be decreased in order to reduce the computational time of the clustering process. Sections 5, 6 and 7 introduce the clustering algorithms that we have developed at OSU including a comparison of the major clustering algorithms applied to scenario analysis. In Section 8, we present some of the results obtained by the clustering algorithms for different types of data sets and highlight the applications of these methodologies in a safety analysis environment.

2 SCENARIO ANALYSIS

The data generated by dynamic methodologies (Siu 1994) such as the dynamic event tree (DET) methodology (Cojazzi 1996) for the analysis of nuclear power plants are usually inhomogeneous due to the fact that they contain

- the temporal descriptions of the state variables of each node of the simulator (e.g., temperature, pressure, level or concentration of particular elements), and,
- the status of system components, both hardware and software (e.g., aperture of a valve or status of a digital control system), and sub-systems (e.g., Emergency Core Cooling System) of the plant under consideration

While the former data type is generally continuous, the latter is typically discrete. When dealing with nuclear transients, it is possible to group the set of scenarios into two possible modes:

- *End State Analysis* which groups scenarios into clusters based on the end state of the scenarios (e.g., NUREG-1150 (U.S.NRC 1990))
- *Transient Analysis* which groups scenarios into clusters based on their time evolution (Mandelli, Aldemir, Yilmaz, Metzroth, & Denning 2010b)

Moreover, it is possible to characterize each scenario based on

- the status of a set of components (Zio & Baraldi 2005), and,
- the temporal behavior of a set of state variables (Mandelli, Aldemir, Yilmaz, Metzroth, & Denning 2010b) (e.g., node pressure, temperature)

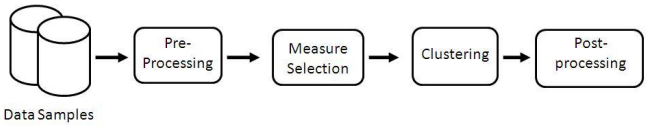

Figure 1: The flow chart for a clustering process.

Our research activity is focused on the latter type of scenario characterization (Mandelli, Aldemir, Yilmaz, Metzroth, & Denning 2010b). Figure 1 gives an overview of the major steps involved in the analysis of the scenarios generated by dynamic methodologies:

Pre-processing. The user chooses how each is scenario is being characterized. Dimensionality reduction and data normalization are also performed (see Section 4).

Measure Selection. Similarity measure (e.g., distance metrics), along with other clustering parameters, are chosen (see Section 5).

Clustering. Data are partitioned into clusters according to the chosen distance metric. Cluster centers and cluster memberships are determined (see Section 6).

Post-Processing. Each cluster is characterized by analyzing sequencing and timing of events for all the scenarios contained in it (see Section 8).

3 DATA REPRESENTATION

Since the temporal evolution of each scenario is typically described by the time evolution of all system state variables (e.g., pressure and temperature at a computational node), we chose to represent each scenario $\vec{x_i}$ $(i = 1, \ldots, I)$ by M state variables x_{im} $(m = 1, \ldots, M)$ plus time t (ranging from 0 to T) as the state vector

$$\begin{aligned} \vec{x_i} \;=\; & [x_{i1}(t_1), \ldots, x_{iM}(t_1), \\ & \ldots, x_{i1}(t_K), \ldots, x_{iM}(t_K)] \end{aligned} \tag{1}$$

where $x_{im}(t_k)$ corresponds to the value of the variable x_m (e.g., temperature, pressure at a computational node) sampled at time t_k (e.g., $t_1 = 0$ and $t_K = T$) for scenario i. Note that the dimensionality of each scenario is $M \cdot K$ and can be extremely high for complex systems (i.e., large number of state variables and large number of samples).

The variables of interest may be chosen a priori by the user depending on which phenomena the user is looking for. Alternatively, depending on the complexity of the system, the user can chose all the state variables. In both cases, we investigated

the possibility of using dimensionality reduction algorithms in order to reduce the number of variables x_m by analyzing their correlation[2] (see Section 4).

It is worth highlighting that the chosen representation gives the flexibility of including new information other than the state variables to characterize each scenario. New information can be included by simply adding new dimensions to the vector shown in Eq. (1). These new dimensions can include

- timing of Q events t_q $(q = 1, \ldots, Q)$
- status of R components c_r $(r = 1, \ldots, R)$, and,
- scenario probability p_i

as shown in Eq. (2)

$$\begin{aligned} \vec{x_i} \;=\; & [x_{i1}(t_1), \ldots, x_{iM}(t_1), \\ & \ldots, x_{i1}(t_K), \ldots, x_{iM}(t_K), \\ & t_1, \ldots, t_Q, c_1, \ldots, c_R, p_i]. \end{aligned} \tag{2}$$

4 DATA PRE-PROCESSING

Dimensionality reduction is the process of finding a bijective mapping function $\mathfrak{F}$

$$\mathfrak{F} : \mathbb{R}^D \mapsto \mathbb{R}^d \text{ (where } d < D) \tag{3}$$

which maps the data points from the D-dimensional space into a reduced d-dimensional space (i.e. embedding on a manifold) in such a way that the distances between each point and its neighbors are preserved. In our applications $D = M + 1$, i.e. M state variables plus time t.

A classical example of manifold analysis is the Swiss-roll (see Fig. 2) which can be studied by only using non-linear algorithms. In this case, points are distributed in a 3-dimensional space (i.e., $D = 3$) which are actually lying on a 2-dimensional space (i.e., $d = 2$). The manifold in this case is represented by a 2-dimensional plane. In our approach, we tackled this dimensionality reduction problem in two possible ways:

Manifold Analysis. This series of algorithms can identify subspaces around each point by generating a graph around its neighbors as in the ISOMAP (Tenenbaum, de Silva, & Langford 2000).

Local Principal Component Analysis (PCA). This methodology relies on the local application of PCA, which assumes local linear correlation among variables, to model a global non-linear correlation (Jolliffe 2002).

[2]Correlations originated from the balance (e.g., mass or energy) and the state equations (e.g., gas state equation)

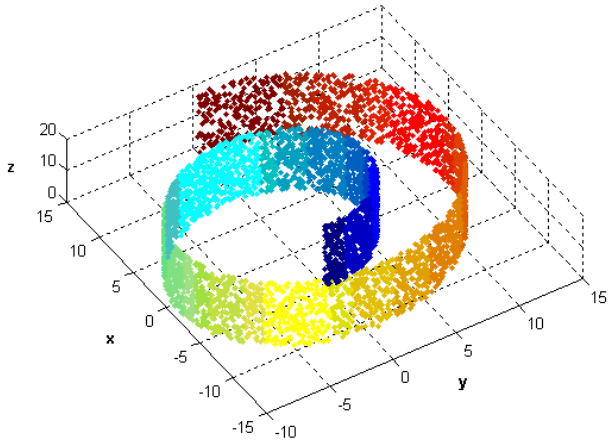

Figure 2: Swiss-roll: example of a 2-dimensional manifold ($d = 2$) in a 3-dimensional space ($D = 3$).

For dimensionality reduction, we first implemented the ISOMAP algorithm and applied it to a dataset[3] (Mandelli, Yilmaz, & Aldemir 2011a) composed by a series of 104 scenarios each one represented by eight state variables of interest[4] (i.e., $M = 8$ and $D = 9$). In (Mandelli, Yilmaz, & Aldemir 2011a), we show that it is possible reduce the number of dimensions from $D = 9$ to $d = 6$ without losing accuracy in the clustering process.

Classical PCA (Jolliffe 2002) can model only linear dependencies among variables using the *eigenspace decomposition*. For complex systems, state variables are related thorough non-linear relationships. However the local application of the PCA algorithm shows promising results for the dimensionality reduction problem as shown in (Mandelli, Yilmaz, & Aldemir 2011c).

Once each scenario is characterized using (1), data normalization is often required due to the fact that the state variables are different in nature and consequently, in their range. This problem can be solved in two ways:

- Normalize each dimension into the $[0, 1]$ interval
- Normalize each dimension by dividing it by its standard-deviation.

[3]The initiating event investigated is a station blackout (SBO) at a U.S. PWR and the MELCOR code is linked to the ADAPT tool to determine the evolution for each DET scenario. The simulations using the MELCOR model of the transient from the occurrence of the SBO through the core melting phase and up to point of containment failure and release of radionuclides to the environment.

[4]Seal LOCA flow rate, hydrogen mass generated, core water level, system pressure, core vapor temperature, hot leg vapor temperature, intact core fraction and fuel temperature.

Table 1: Summary of the commonly used measures.

Measure	Form
Minkowski	$d_n(\vec{x_i}, \vec{x_j}) = (\sum_{r=1}^{M \cdot K} \lvert\vec{x_i}(r) - \vec{x_j}(r)\rvert^n)^{\frac{1}{n}}$
Euclidean	$d_2(\vec{x_i}, \vec{x_j}) = (\sum_{r=1}^{M \cdot K} \lvert\vec{x_i}(r) - \vec{x_j}(r)\rvert^2)^{\frac{1}{2}}$
Taxicab	$d_1(\vec{x_i}, \vec{x_j}) = \sum_{r=1}^{M \cdot K} \lvert\vec{x_i}(r) - \vec{x_j}(r)\rvert$
Supremum	$d_0(\vec{x_i}, \vec{x_j}) = max_r \lvert\vec{x_i}(r) - \vec{x_j}(r)\rvert$
Mahalanobis	$d_M(\vec{x_i}, \vec{x_j}) = (\vec{x_i} - \vec{x_j})^T S^{-1} (\vec{x_i} - \vec{x_j})$

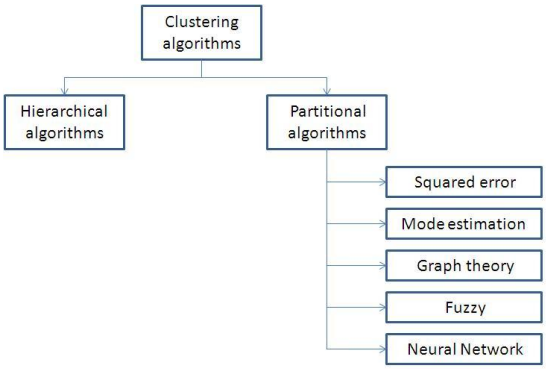

Figure 3: Taxonomy of clustering methodologies.

5 MEASURES

A cluster is a collection of objects which are similar to each other and are dissimilar to the objects belonging to other clusters (Rui & Ii 2005). Common measures of similarities (or, dissimilarities) which are used in this article are distances. In the literature (Mendelson 1990), it is possible to find several types of distances other than the Euclidean distance and its more general formulation (i.e., the Minkowski distance) as shown in Table 1. The approach of using distance metrics to clustering is called distance-based clustering which is used in the following discussion.

In most of our past work, we have used the Euclidean distance. Clustering by using different type of metrics is still part of our ongoing research.

6 CLUSTERING ALGORITHMS

From a mathematical viewpoint, the concept of clustering (Rui & Ii 2005) that we aim is to find a partition $\mathbf{C} = \{C_1, \ldots, C_l, \ldots, C_L\}$ of the set of I scenarios $\mathbf{X} = \{\vec{x_1}, \ldots, \vec{x_i}, \ldots, \vec{x_I}\}$ where each scenario $\vec{x_i}$ is represented as a multi-dimensional vector as shown in (1). Each C_l $(l = 1, \ldots, L)$ is called a cluster. The partition $\mathbf{C}$ of $\mathbf{X}$ is given as follows:

$$\begin{cases} \mathbf{C}_l \neq \varnothing, l = 1, \ldots, L \\ \bigcup_{l=1}^{L} \mathbf{C}_l = \mathbf{X} \end{cases} \tag{4}$$

As shown in Fig. 3 (Jain, Dubes, & Richard 1988), the main division between clustering methodologies can be made by partitioning them into two classes (Rui & Ii 2005):

- Hierarchical algorithms
- Partitional algorithms

Hierarchical algorithms build a hierarchical tree from the individual point (leaf) by progressively merging them into clusters until all points are inside a single cluster (root). Partitional clustering, on the other hand, seeks a single partition of the data sets instead of a nested sequence of partitions obtained by hierarchical methodologies. Under this category it is possible to classify methodologies under five main sub-categories: Squared Error (e.g., K-Means (MacQueen 1967)), Fuzzy clustering (e.g., Fuzzy C-Means (Bezdek 1981)), Mode Seeking (e.g., Mean-Shift (Fukunaga & Hostetler 1975)), Graph Theoretical (van Groenewoud 1974) and Neural Network based (Wei, Su, Qiu, Ni, & Yang 2010).

We initially evaluated hierarchical algorithms as well as partitional algorithms including (Mandelli, Yilmaz, & Aldemir 2011b):

- Squared Error
- Fuzzy C-Means
- Mode-Seeking

Hierarchical algorithms (Duda, Hart, & Stork 2000) organize data into a hierarchical structure accordingly to a proximity matrix in which an entry (α, β) is some measure of the similarity (or distance) between the items to which row α and column β corresponds. Usually, the final result of these algorithms is a binary tree, also called dendrogram, in which the root of the tree represents the whole dataset and each leaf is a data point.

Squared Error (Duda, Hart, & Stork 2000) algorithms assign each point to a cluster whose center (also called centroid) is nearest. The cluster center is the average of all the points in the cluster, that is, its coordinates are the arithmetic mean of each dimension independently over all the points in the same cluster. The most famous and used methodology is the K-Means algorithm (MacQueen 1967), where the user specifies a priori the number of clusters to be determined.

Fuzzy C-Means (Bezdek 1981) clustering is very similar to K-Means but it assigns a degree of belonging $u_l(\vec{x_i})$ for each point $\vec{x_i}$ to each cluster l, as in fuzzy logic, rather than assigning it to a single cluster.

Mode-Seeking (Cheng 1995) is based on the assumption that the distribution of the points in the state space can be described through an unknown probability density function (pdf). The goal is to find the modes with highest probability, i.e. the regions in the state space with higher data densities. In this case, the number of obtained clusters is dependent on local density function that are used. A particular mode seeking approach the Mean-Shift methodology (Fukunaga & Hostetler 1975), which has been used for a number of applications in different fields.

For our purpose, we identify Mean-Shift algorithm as the most promising approach for the following reasons:

- Hierarchical, K-Means and Fuzzy C-Means algorithms are able to identify clusters of points having only spherical or ellipsoidal shape while Mean-Shift can identify clusters having any arbitrary geometry.

- Hierarchical, K-Means and Fuzzy C-Means algorithms have difficulty identifying outliers, i.e., clusters having a very small number of points in it. On the other side, Mean-Shift can easily identify scenarios that are considerably distant from the others.

- The level of discrimination between the clusters could be specified using the bandwidth parameter of the Mean-Shift algorithm. In that way, the appropriate number of clusters is determined by the algorithm itself instead of specifying the number of clusters that are going to be determined as required by K-Means and Fuzzy C-Means algorithms.

7 MEAN-SHIFT ALGORITHM

Mean-Shift algorithm (Fukunaga & Hostetler 1975) is a non-parametric iterative procedure that shifts each data point to the average of data points in its neighborhood in order to determine the cluster centers and to assign each point to one cluster center only. By cluster center we mean a region with high observation density (i.e., the modes of the dataset).

The main idea is to consider each point $\vec{x_i}$ ($i = 1, \ldots, I$) of the dataset as an empirical distribution density function, or kernel, $K(\vec{x})$ distributed in a multidimensional space where regions with high data density (i.e., modes) correspond to local maxima of the multivariate kernel density estimate $f_I(\vec{x})$ (Cacoullos 1966) (see Fig. 4) defined as:

$$f_I(\vec{x}) = \frac{1}{Ih^d} \sum_{i=1}^{I} K\left(\frac{\vec{x} - \vec{x_i}}{h}\right) \tag{5}$$

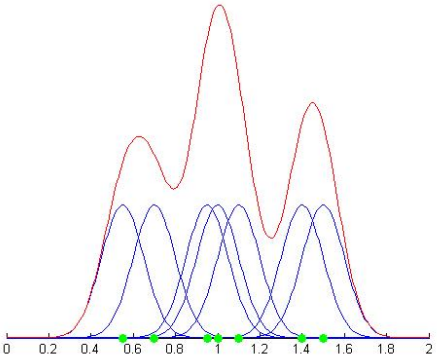

Figure 4: Density function (red line) for points distributed in a 1-dimensional space modeled using kernels (blue lines)

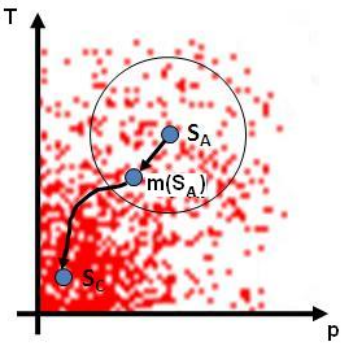

Figure 5: Determination of a cluster center in a 2-dimensional space using a Mean-Shift algorithm.

where each element $\vec{x} \in \mathbb{R}^{M \cdot K}$[5].

In order to determine the points with high data density (i.e., the modes of $f_I(\vec{x})$), we are interested in the solution of $\nabla f_I(\vec{x}) = 0$.

Starting from an arbitrary point (e.g., point S_A in Fig. 5), the algorithm associates a hypersphere (depending on the number of dimensions of the state space) centered at that point with radius equal to h. The objective is to consider all the points that are inside the hypersphere and determine the center of mass[6] of these points from the point $m(S_A)$ (see Fig. 5 for illustration):

$$m(s_A) = \frac{\sum_{s \in S} K(s - s_A)s}{\sum_{s \in S} K(s - s_A)} \tag{6}$$

where the function $K(x)$ is the kernel chosen to model the local distribution.

The algorithm then moves from the original point S_A in Fig. 5 into the calculated position $m(s_A)$ and repeatedly computes the center of mass for the

[5]Note that it is possible to perform a probabilistic clustering by using the probability as a weighting factor. Each scenario has a different weight w_i proportional to its own probability and $f_I(\vec{x})$ can be rewritten in this form: $f_I(\vec{x}) = \frac{1}{Ih^d} \sum_{i=1}^{I} w_i K\left(\frac{\vec{x} - \vec{x_i}}{h}\right)$

[6]The center of mass of a finite set of points is a weighted average position of these points in the space.

points included inside the hypersphere but now centered on $m(S_A)$. This operation converges to the mode when the distance between the new center of mass and the old one is below a fixed threshold (point S_C in Fig. 5). When this condition is reached:

- point S_C is considered the center of a cluster
- the original point S_A is uniquely associated to the cluster centered by point S_C

Repeating this process for all the points in the dataset provides:

- the center of all the clusters and the list of all the points that belong to that specific cluster
- the cluster to which each point belongs (as mentioned, each point belongs to one cluster only)

An important issue about the application of clustering algorithm is the computational time required to analyze large data sets generated by DET. Typical sizes are of the order of megabytes or gigabytes which may require several hours of computation for a single value of bandwidth. Thus, we implement the algorithm in a parallel fashion in `C++` using parallel directives of OpenMP (Chandra 2001). Due to the nature of the algorithm, this step did not require major changes in the original structure of the code. In particular, the analysis of the membership of each point to a cluster, through the sequence of computations of center of mass, is divided into threads (one thread for the computation of a single point). The choice of OpenMP is primarily due to the fact that each thread requires the full dataset and, hence, a set of shared memory directives (e.g., OpenMP) is required.

8 APPLICATIONS

We applied the Mean-Shift algorithm on several data sets. The first dataset is generated by a DET algorithm for the analysis of the controller of heated water tank (Mandelli, Aldemir, Yilmaz, & Denning 2010). This dataset contains a set of 619 scenarios where each one is described by two variables (temperature and level) sampled 200 times over a period of four hours. Clustering results are shown in (Mandelli, Aldemir, Yilmaz, & Denning 2010). The other data sets we consider are larger data sets as introduced in the following sections.

8.1 *ABR1000 Analysis*

This data set is more complex and is generated by ADAPT (Rutt, Catalyurek, Hakobyan, Metzroth, Aldemir, Denning, Dunagan, & Kunsman 2006) for the analysis of an aircraft crash scenario

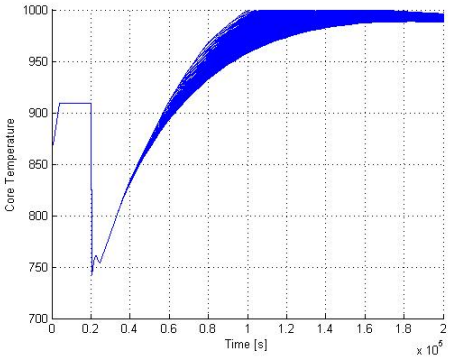

Figure 6: Graphical representation of the scenarios generated by ADAPT for the aircraft crash scenario.

of a ABR1000 reactor (Kim, Yang, Grandy, & Hill 2008). ADAPT is a DET generation methodology coupled to a dynamic model of the system under consideration (Rutt, Catalyurek, Hakobyan, Metzroth, Aldemir, Denning, Dunagan, & Kunsman 2006). The dataset is generated by an initiating event in the form of an aircraft crashing into the plant at time zero with the plant operating at 100% power (Winningham, Metzroth, Aldemir, & Denning 2009). Three of the four towers are assumed to be destroyed, producing debris that blocks the air passages (hence, impeding the possibility to remove the decay heat). The reactor trips, offsite power is lost, the pump trips and coasts down. A recovery crew and heavy equipment are used to remove the debris from each tower once at at time. The analysis has been carried out using ADAPT coupled with RELAP5 (RELAP5-3D Code Development Team 2005) as the system simulator; the generated dataset contains 610 different scenarios as illustrated in Fig. 6.

In this scenario, the processed data consists of the temporal description of the only one variable chosen (maximum temperature of the core) and the timing of the events such as the arrival of the recovery crew and the recovery of the three towers. We sampled the maximum temperature of the core at 56 time points. In (Mandelli, Aldemir, Yilmaz, Metzroth, & Denning 2010a) we performed the clustering using Mean-Shift methodology. Figure 7 shows the obtained cluster centers that correspond to the representative scenarios.

We use the timings of crew arrival and tower recovery to characterize each scenario. Figure 8 shows these quantities for Cluster 1. Figure 8(a) shows a comparison between the cluster center and the scenarios that belong to it. Figure 8(b) presents a histogram which shows the timing of events (crew arrival time and tower recovery) for member scenarios.

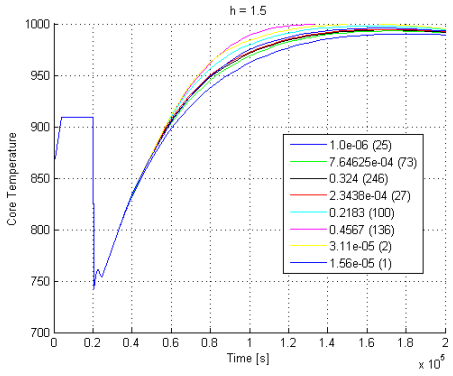

Figure 7: Cluster centers for the RVACS system for $h = 1.5$. The numbers in the legend indicate the fraction of scenarios that fall in each cluster. The numbers in the parenthesis show the number of scenarios in the cluster

8.2 *Pump Seal Leakage Model*

We evaluate the difference between three different datasets generated by ADAPT for Zion plant during an"offsite power loss" scenario. The differences between the three system configurations are the models implemented to describe the pump seal leakage.

We performed the clustering of the three datasets separately. Figures 9(a), 9(b) and 9(c) show the clusters corresponding to each of the three pump seal leakage models where the primary system pressure is used as the scenario descriptor. The lines in Fig. 9 denote the estimated cluster centers. Figure 9 also shows the shaded regions around the cluster center which indicate the member scenarios contained in them. These shaded regions spread around the cluster centers in a manner analogous to the error bars and their width indicates how the scenarios contained in that cluster are spread in the state space.

From Fig. 9 we identify the following:

- Cluster 1: Figures 9(a), 9(b) and 9(c) have this cluster in common which is composed of a single scenario
- Cluster 2: Figures 9(a), 9(b) and 9(c) have in common this cluster but in Fig. 9(b) the shaded bar is narrower
- Clusters 3 and 4: Figures 9(a), 9(b) and 9(c) have in common these clusters composed by a single scenario.
- Cluster 5: This cluster is in common in Fig. 9(a) and Fig. 9(c) while is not present in case (b).

As a last remark, it is possible to note that in the region marked as 6, Fig. 9(a) and Fig. 9(c) are very similar while in Fig. 9(b) there is only one cluster scenario which includes scenarios that after 6000 s are characterized by stable system pressure at $16 \cdot 10^6$ Pa.

9 CONCLUSIONS

We have presented a summary of the research carried out at OSU for the analysis of a large number of data sets generated by safety analysis codes. Each scenario contains information about the temporal behavior of the state variables, status of the components/systems, and timing/sequence of the events. We focused our attention on the major steps involved in the analysis: pre-processing, measure selection, data clustering and cluster analysis. We then showed some relevant applications.

We illustrated how the data can be grouped into clusters depending on their temporal evolution. The estimated cluster centers provide the most relevant trends of the overall analysis. The grouping of scenarios into clusters is shown to be helpful for identifying such trends and evaluating their characteristics. Such groupings are also valuable to identify the differences between data sets generated for different system configurations.

REFERENCES

Aldemir, T., U. Catalyurek, R. Denning, C. Smidts, X. Sun, & A. Yilmaz (2011). Method and tool development to support systematic quantification of uncertainties. *to appear in Trans. Am. Nucl. Soc 104*.

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Norwell, MA, USA: Kluwer Academic Publishers.

Bishop, C. M. (2007). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer.

Cacoullos, T. (1966). Estimation of a multivariate density. *Annals of the Institute of Statistical Mathematics 18*(1), 179--189.

Chandra, R. (2001). *Parallel programming in OpenMP*. Morgan Kaufmann Publishers Inc.

Cheng, Y. (1995). Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence 17*(8), 790--799.

Cojazzi, G. (1996). The dylam approach for the dynamic reliability analysis of systems. *Reliability Engineering and System Safety 52*(52), 279--296.

Duda, R., P. Hart, & D. Stork (2000). *Pattern Classification*. Wiley-Interscience Publication.

Fukunaga, K. & L. Hostetler (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory 21*(1), 32--40.

Jain, A. K., K. Dubes, & C. Richard (1988). *Algorithms for clustering data*. Upper Saddle River, NJ (USA): Prentice-Hall, Inc.

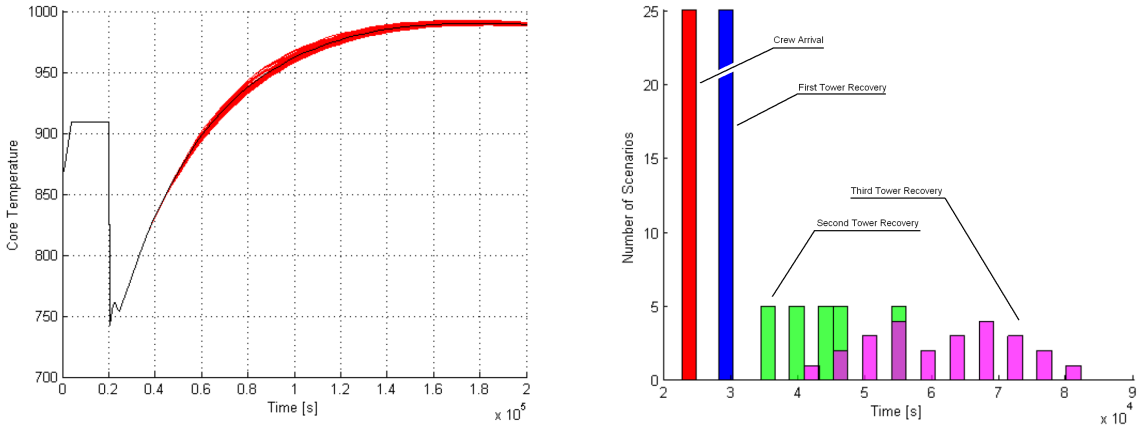

Figure 8: Analysis of Cluster 1. Left: Cluster center (black line) and scenarios belonging to to Cluster 1 (red lines). Right: histogram of crew arrival time (red) and tower recovery (blue, green and magenta) for the scenario belonging to Cluster 1.

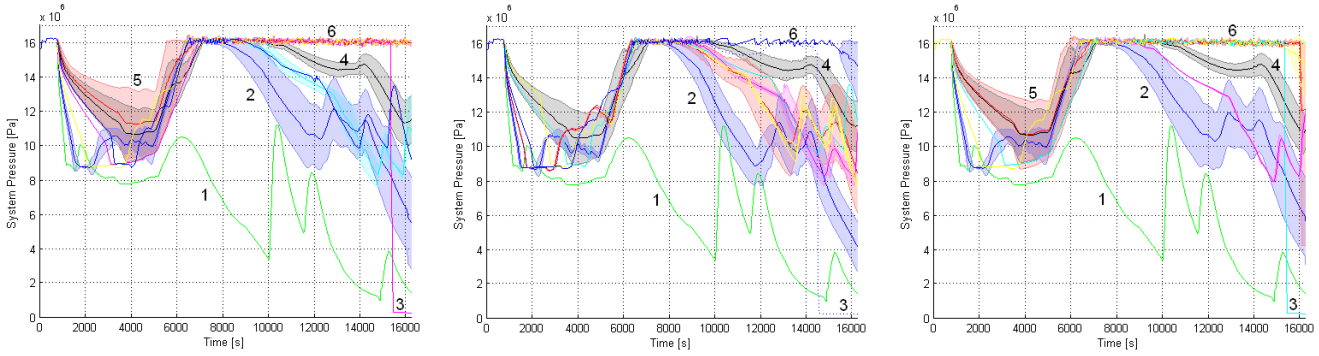

Figure 9: Clusters obtained from 3 different models of pump seal leakage. Lines denote cluster centers, shaded region indicates how the scenarios contained spread around the cluster center. Numbers denote clusters identifiers.

Jolliffe, I. T. (2002, October). *Principal Component Analysis* (Second ed.). Springer.

Kim, T. K., W. S. Yang, C. Grandy, & R. N. Hill (2008, September). Core design studies for a 1000 MWt advanced burner reactor. In *Proceedings of PHYSOR 2008, Interlaken (Switzerland)*.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In L. M. L. Cam and J. Neyman (Eds.), *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, Volume 1, pp. 281--297. University of California Press.

Mandelli, D., T. Aldemir, A. Yilmaz, & R. Denning (2010). Scenario aggregation and analysis via mean-shift methodology. In *Proceeding on Probabilistic Safety Assement and Management (PSAM10)*.

Mandelli, D., T. Aldemir, A. Yilmaz, K. Metzroth, & R. Denning (2010a). Scenario aggregation and analysis via mean-shift methodology in level 2 PRA. In *Proceedings of the American Nuclear Society (ANS) ICAPP 2010 Topical Meeting International Congress on Advances in Nuclear Power Plants*, pp. 990--994.

Mandelli, D., T. Aldemir, A. Yilmaz, K. Metzroth, & R. Denning (2010b). Scenario aggregation in dynamic probabilistic risk assessment. *Draft for Reliability Engineering and System Safety*.

Mandelli, D., A. Yilmaz, & T. Aldemir (2011a). Clustering scenarios on manifolds. In *Proceeding of American Nuclear Society (ANS)*.

Mandelli, D., A. Yilmaz, & T. Aldemir (2011b). Data processing methodologies applied to dynamic PRA: an overview. In *Draft accepted for Topical meetings on Probabilistic Safety Analysis (PSA)*.

Mandelli, D., A. Yilmaz, & T. Aldemir (2011c). Dimensionality reduction using local pca. In *Proceeding of American Nuclear Society (ANS)*.

Mendelson, B. (1990). *Introduction to Topology*. New York (NY), USA: Dover Publications.

Mercurio, D., L. Podofillini, E. Zio, & V. Dang (2009). Identification and classification of dynamic event tree scenarios via possibilistic clustering: Application to a steam generator tube rupture event. *Accident Analysis and Prevention 41*, 11801191.

RELAP5-3D Code Development Team (2005). *RELAP5-3D Code Manual*. Idaho National Laboratory, Idaho Falls, ID (USA).

Rui, X. & Ii (2005, May). Survey of clustering algorithms. *IEEE Transactions on Neural Networks 16*(3), 645--678.

Rutt, B., U. Catalyurek, A. Hakobyan, K. Metzroth, T. Aldemir, R. Denning, S. Dunagan, & D. Kunsman (2006). Distributed dynamic event tree generation for reliability and risk assessment. In *Challenges of Large Applications in Distributed Environments*, pp. 61--70. IEEE.

Siu, N. (1994). Risk assessment for dynamic systems: an overview. *Reliability Engineering and System Safety 43*(1), 43--73.

Tenenbaum, J. B., V. de Silva, & J. C. Langford (2000). A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 2319--2323.

U.S.NRC (1990). *NUREG 1150 - Severe accident risks: an assessment for five U.S. nuclear power plants*. Division of Systems Research, Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, Washington, DC.

van Groenewoud, H. (1974). A cluster analysis based on graph theory. *Plant Ecology 29*(2), 115--120.

Wei, H., G. Su, S. Qiu, W. Ni, & X. Yang (2010). Applications of genetic neural network for prediction of critical heat flux. *International Journal of Thermal Sciences 49*(1), 143 -- 152.

Winningham, R., K. Metzroth, T. Aldemir, & R. Denning (2009). Passive heat removal system recovery following an aircraft crash using dynamic event tree analysis. In *Proceeding of American Nuclear Society (ANS)*, Volume 100, pp. 461--462.

Zio, E. & P. Baraldi (2005). Identification of nuclear transients via optimized fuzzy clustering. *Annals of Nuclear Energy 32*(10), 1068 -- 1080.