CoDDA: A Flexible Copula-based Distribution Driven Analysis Framework for Large-Scale Multivariate Data

Subhashis Hazarika, Soumya Dutta, Han-Wei Shen, *Member, IEEE*, and Jen-Ping Chen

Abstract—CoDDA (**Co**pula-based **D**istribution **D**riven **A**nalysis) is a flexible framework for large-scale multivariate datasets. A common strategy to deal with large-scale scientific simulation data is to partition the simulation domain and create statistical data summaries. Instead of storing the high-resolution raw data from the simulation, storing the compact statistical data summaries results in reduced storage overhead and alleviated I/O bottleneck. Such summaries, often represented in the form of statistical probability distributions, can serve various post-hoc analysis and visualization tasks. However, for multivariate simulation data using standard multivariate distributions for creating data summaries is not feasible. They are either storage inefficient or are computationally expensive to be estimated in simulation time (*in situ*) for large number of variables. In this work, using copula functions, we propose a flexible multivariate distribution-based data modeling and analysis framework that offers significant data reduction and can be used in an *in situ* environment. The framework also facilitates in storing the associated spatial information along with the multivariate distributions in an efficient representation. Using the proposed multivariate data summaries, we perform various multivariate post-hoc analyses like query-driven visualization and sampling-based visualization. We evaluate our proposed method on multiple real-world multivariate scientific datasets. To demonstrate the efficacy of our framework in an *in situ* environment, we apply it on a large-scale flow simulation.

Index Terms—In situ processing, Distribution-based, Multivariate, Query-driven, Copula

1 Introduction

Scientists often measure multiple physical attributes/variables at the same time in their computational models. These variables are used to perform various multivariate analyses to gain in-depth insights into the underlying physical phenomenon. Recent advances in the field of high-performance computing have enabled scientists to simulate their computational models at very high resolutions, thus, generating data in the scale of terabytes or even petabytes. The multivariate nature of the simulation adds to the complexity of such large-scale scientific datasets, thereby, possessing significant challenges with respect to performing multivariate analysis and visualization tasks.

A popular and effective strategy for analyzing and visualizing large-scale scientific datasets is to first partition the simulation domain and then store statistical data summaries for each partition [12, 14, 17, 32]. This strategy is particularly useful in many *in situ* applications to alleviate issues like storage overhead and I/O bottleneck for large-scale data. Such applications create the data summaries *in situ* (i.e, while the simulation is still running) and write-out the compact statistical representation instead of the raw data. These summarized data representations are later used to perform post-hoc analysis and visualization in a much scalable manner (even on commodity hardware). Such summaries, often represented in the form of various statistical probability distributions (Histogram, Gaussian Mixture Models, etc.) offer two significant benefits. *First*, storing probability distributions for local neighborhood helps reduce the overall storage footprint for large-scale datasets. *Second*, many feature-based and query-driven analysis and visualization tasks rely on computing local data statistics, which makes such statistical summaries a prudent choice for compact data representation [15, 24, 35, 47, 48]. However, for multivariate data, where it is important to preserve the multivariate relationship among variables, using standard multivariate probability distribution models for data summarization does not always yield similar benefits. They are either not space efficient for the purpose of data reduction (e.g multivariate histograms) or are computationally very expensive to estimate when the number of variables increases, thus, overburdening the actual simulation execution (e.g multivariate Gaussian Mixture Models). Therefore, there is a need to rethink how to model large-scale multivariate data, such that we still have similar benefits as univariate data summaries. Moreover, performing multivariate analysis tasks *in situ* may not always be helpful, especially, for exploratory analysis tasks [12], where, in the initial stages scientists usually do not have a clear understanding of the important variables to analyze and/or the precise value ranges to query for [16]. Such exploratory analysis involves back-and-forth interaction with the data, trying various choices before developing a clear idea. However, it is often computationally prohibitive to run large simulations in supercomputing environments multiple times for such exploratory analysis. Therefore, there is a real necessity to have a good multivariate data summarization solution for large-scale multivariate simulations, that can preserve the various multivariate relationships as well as be computationally efficient both with respect to storage footprint and estimation time.

In this paper, we propose a flexible distribution-driven analysis framework for large-scale multivariate data that addresses the aforementioned concerns. In the first stage of our framework, to achieve a compact data representation, we partition the simulation domain and store the corresponding univariate distributions of the variables for each partition. The dependency among the variables for each partition is separately estimated using copula functions. Copula functions offer a statistically robust mechanism to model the dependency structures of variables irrespective of the type of univariate distributions used to model the individual variables. As a result of this flexibility, they have been widely used in the field of financial modeling [11, 19, 40], machine learning [18, 31, 49, 55] and recently, in the field of visualization, for uncertainty modeling in ensemble datasets [27]. To preserve the spatial information in our model, we also consider the spatial variables as extra dimensions along with the physical variables and store the corresponding spatial distributions in an efficient representation. In the second stage of our framework, to demonstrate the efficacy of our proposed multivariate data representation, we perform two broad categories of post-hoc multivariate analysis tasks using a copula-based sampling strategy. (a) For effective post-hoc visualization, we propose a multivariate sampling-based technique to create sample scalar fields of arbitrary user-specified grid resolutions. (b) For multivariate query-driven analysis tasks, we propose the computation of probabilis-

- *Subhashis Hazarika, Soumya Dutta, and Han-Wei Shen are with the GRAVITY research group, The Department of Computer Science and Engineering, The Ohio State University. E-mail: hazarika.3, dutta.33, shen.94@osu.edu.*
- *Jen-Ping Chen is with The Department of Mechanical and Aerospace Engineering, The Ohio State University. E-mail: chen.1210@osu.edu.*

Manuscript received 31 Mar. 2018; accepted 1 Aug. 2018.
Date of publication 16 Aug. 2018; date of current version 21 Oct. 2018.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2018.2864801

tic multivariate queries from our data summaries. Besides evaluating our proposed data modeling strategy on two large-scale multivariate datasets, we also test our method in a real-world *in situ* scenario, by running it directly with a large-scale CFD simulation. We conduct both quantitative and qualitative assessment of our generated results and offer insights into various choices that we make.

To summarize, the major contribution of our work is twofold:

- To reduce the overall storage footprint of large-scale multivariate data, we propose a statistically robust strategy to model multivariate distributions, which is computationally efficient to be run *in situ* during the simulation execution time.

- To perform efficient post-hoc visualization and exploration of multivariate data, we propose a copula-based sampling strategy to generate spatial-context preserving sample scalar fields as well as facilitate query-driven analysis by computing probabilistic multivariate queries from our proposed data summaries.

2 Related Work

In this section, we focus on some of the previous works related to the ideas behind our proposed framework.

Distribution-Driven Analysis: Statistical probability distributions have been widely used in the field of scientific data analysis and visualization [30, 36, 47, 48]. Liu et al. [33] exploited GMMs for stochastic sampling-based volume rendering on the GPU. Lundstrom et al. [35] studied the design of transfer functions in direct volume rendering based on local histograms. Distributions have also been widely used to model uncertainty in scientific datasets. Jarema et al. [29] used directional distributions to perform comparative visual analysis of vector field ensembles. Several methods have been proposed to visualize and extract uncertain features like isosurfaces [2, 43–46], vortices [27, 41] and streamlines [21] from distribution fields. With respect to distribution-based data summarization for large-scale data, Thompson et al. [56] proposed Hixels, which stores histogram per data block to preserve the statistical properties of data. Dutta et al. [14, 15] stored GMMs per data block to track time-varying uncertain features. Recently, they also proposed homogeneity preserving data partitioning scheme [17], where the local data was modeled using a hybrid mixture of Gaussian distributions and GMMs. Wang et al. [58] stored spatial GMMs per bin of the local data histogram to achieve good reconstruction results. Almost all of these distribution-based data summarization works are targeted for univariate dataset. In this work, we proposed a framework to facilitate distribution-based data summarization for large-scale multivariate data.

Multivariate Analysis: Multivariate analysis and visualization is a well-researched topic in the field of scientific visualization [22, 60]. Sauber et al. [52] studied the local correlation coefficients among the variables to analyze and visualize multivariate data. Bethel et al. [4] computed correlation fields to perform query-driven analysis with multivariate data. Gosnik et al [24] used local statistical distributions to improve query-driven analysis for multivariate data. Jänicke et al. [28] adapted local statistical complexity to identify informative regions in multivariate data. Creating efficient multivariate distributions have always been a challenging task. Various compact representations of the multivariate joint histogram have been proposed to tackle the curse of dimensionality [6, 34].

***In situ* Application:** With increasing sizes of scientific simulation data, *in situ* data processing is becoming increasingly popular for scalable analysis and visualization tasks. Bauer et al. [3] performed a comprehensive survey of the *in situ* visualization techniques. Direct visualization of the simulation data can be performed with LibSim using VisIt [59] and CATALYST using Paraview [20]. Vishwanath et al. [57] in their work, GLEAN, improved the process of *in situ* analysis. Yu et al. [64] performed *in situ* visualization of combustion data. Woodring et al. [62] proposed an *in situ* eddy census for ocean simulation models. However, exploratory data analysis tasks, which require back-and-forth interaction with the raw data are not feasible with pure *in situ* techniques [16]. To address such limitations, recently, a new *in situ* practice has been gaining popularity, where, large-scale data is statistically summarized and later used for post-hoc analysis using the data summaries rather than the raw data [12, 32]. An *in situ* image-based approach was used by Ahrens et al. [1] for post-hoc feature exploration. Woodring et al. [61] adopted a sampling-based method to visualize Cosmology data. To facilitate interactive post-hoc visualization of particle data, Ye et al. [63] computed probability distribution functions *in situ*. Dutta et al. [14, 17] performed *in situ* estimation of combinations of GMMs to create data summaries, which are later used for post-hoc feature exploration. To the best of our knowledge, similar approaches to facilitate post-hoc multivariate analysis on large-scale multivariate data does not exist. In this paper, we propose a new multivariate distribution-based data modeling strategy to address this scenario.

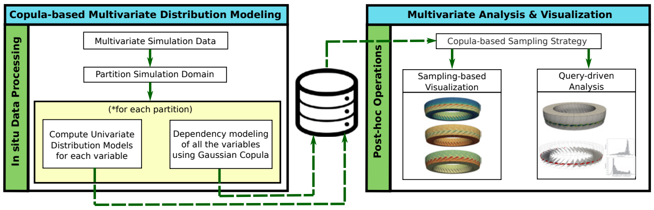

Copula-based Multivariate Distribution Modeling
In situ Data Processing
Multivariate Simulation Data
Partition Simulation Domain
(*for each partition)
Compute Univariate Distribution Models for each variable
Dependency modeling of all the variables using Gaussian Copula
Multivariate Analysis & Visualization
Post-hoc Operations
Copula-based Sampling Strategy
Sampling-based Visualization
Query-driven Analysis

Fig. 1: A schematic overview of the stages of our proposed method.

Copula-based Statistical Analysis: The relationship between a generic multivariate function and a copula function was first formalized by Sklar in 1959 [54]. Since then it has been widely used as a robust statistical tool for multivariate data modeling. In the article titled, *Coping with Copula* [53], Schmidt provides a detailed explanation of the workings of coupla functions and their potential application in various fields. Copula functions have been widely used in the field of financial modeling and risk analysis [11, 19, 39, 40]. Over the past few years, copula functions, especially, Gaussian copula, have been gaining popularity in the field of machine learning as well, for the purpose of modeling high-dimensional distributions [18, 49]. Machine learning approaches like dimensionality reduction [25, 26], mixture modeling [23, 55], component analysis [31, 37] and clustering [50] have benefited from the flexibility offered by copula functions. Recently, in the field of visualization, Hazarika et al. [27] used Gaussian copula functions to model the local neighborhood uncertainty in ensemble datasets with mixed distribution models. Using their copula-based strategy, they visualized uncertain features like isosurfaces and vortices in ensemble datasets. In our proposed multivariate data summarization framework, we use Gaussian copula function to tackle the challenges of scalable multivariate analysis and visualization in large-scale simulation data.

3 System Overview and Motivation

Overview: Figure 1 provides a schematic overview of the different stages of our proposed framework. The two main stages are: (a) *data modeling/summarization*, which can be performed *in situ* alongside the simulation and (b) subsequent *post-hoc multivariate analysis* using the constructed data summaries. The data modeling stage consists of first partitioning the simulation domain and then modeling the individual variables in each partition using suitable univariate distribution models. The dependency among the variables is modeled separately using copula functions. The dependency parameters and the respective univariate distributions, computed *in situ*, together comprises our proposed multivariate data summary, which gets written-out to the secondary storage instead of the raw simulation data. In the latter stage, copula-based sampling strategies are used to facilitate various post-hoc multivariate analysis and visualization tasks using the stored data summaries.

Motivation: Distribution-based data summarization is an effective strategy for dealing with large-scale scientific data. Because of their compact representations, statistical distributions like Histograms, Gaussian Mixture Models (GMM) and Gaussian distributions are commonly used for this purpose, as compared to less compact models like Kernel Density Estimates (KDE). However, it becomes increasingly difficult to work with their corresponding standard multivariate distribution representations when the dimensionality increases. Some potential disadvantages of using standard multivariate distributions for data summarization in large-scale multivariate data can be categorized as follows:

1. **Storage:** The storage footprint of a multivariate histogram can increase exponentially with the number of variables, making them ineffective for data summarization. Although a sparse representation of the multivariate histogram can reduce the exponential storage size, still, compared to the size of the raw data it is not useful for the purpose of data reduction as shown in our evaluations in Section 6. Moreover, the size of such sparse representations is sensitive to how the data is distributed and the number of histogram bins used.

2. **Estimation Time:** GMM is another popular data summarization alternative because of its compact representation and good modeling accuracy. However, the estimation of multivariate GMM using expectation-maximization is computationally very expensive compared to its univariate counterpart. The computation time increases rapidly with the number of variables. Therefore, despite the storage advantages, the high estimation times of multivariate GMMs will overshadow any I/O bottleneck alleviation, making them infeasible for multivariate data summarization in *in situ* applications.

3. **Flexibility:** Standard multivariate distributions are very rigid with respect to the assumptions made about their corresponding univariate distributions. For example, in a multivariate histogram, the individual variables are also histograms (i.e, marginal histograms) and a multivariate GMM with 3 modes always assume that the individual variables are modeled by univariate GMM with 3 modes. However, if a certain variable can be modeled by a simple Gaussian distribution with sufficient confidence, then, by using a Gaussian distribution (which requires storing just two parameters) instead of a distribution with more parameters to store, we can achieve higher levels of data reduction without compromising on quality, as shown by Dutta et al. [17] on univariate data. Such flexibility is not offered implicitly by the standard multivariate distributions.

In order to address the above issues and design an effective multivariate data summarization technique, we propose the use of *copula functions* to model the multivariate distributions rather than using the standard multivariate distribution models. Copula functions offer a statistically robust mechanism to decouple the process of multivariate distribution estimation into two independent task: univariate distribution estimation and dependency modeling [53]. As a result, the exponential cost of storage and/or distribution estimation time can be reduced significantly because we can independently model the individual variables using arbitrary distribution types, while the copula function captures the dependency among them separately.

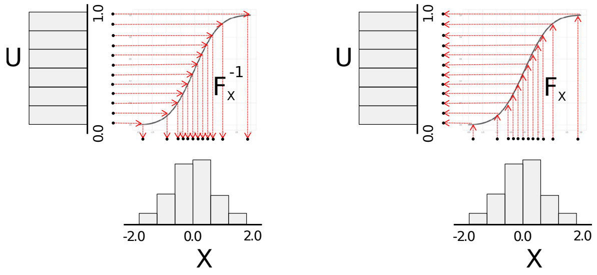
4 Copula-based Multivariate Distribution Modeling

In this section, we explain in detail the first stage of our framework, i.e, multivariate data modeling using copula. We also provided the basic mathematical foundations necessary to understand the application of copula functions for multivariate modeling.

Copula: By definition, a *copula function* or a *copula* in general, is a multivariate cumulative density function (CDF) whose univariate marginals are uniform distributions. Mathematically, $C : [0,1]^d \to [0,1]$ represents a d-dimensional copula (i.e., d-dimensional multivariate CDF) with uniform marginals. For d-uniform random variables $u_1,...u_d$, it can be also be denoted as $C(u_1,...u_d)$.

Sklar's theorem [54] formally established that every joint CDF in $\mathbb{R}^d$ implicitly consists of a d-dimensional copula function. If F is the joint CDF and $F_1,F_2,...F_d$ are the marginal CDF's for a set of d real valued random variables, $X_1,X_2,...X_d$ respectively, then Sklar's theorem can be formally represented as;

$$\begin{aligned} F(x_1,x_2...x_d) &= C(F_1(x_1),F_2(x_2),...F_d(x_d)) \\ &= C(u_1,u_2,...u_d) \quad (\text{using } F_i(x_i)=u_i \ \sim U[0,1]) \end{aligned} \tag{1}$$



(a) $F_X^{-1}(U) \sim X$ (b) $F_X(X) \sim U$

Fig. 2: Property of CDF: (a) If we know the inverse CDF F_X^{-1} of a distribution of variable X, we can always transform uniform samples to follow distribution of X (b) The output of a continuous CDF F_X, is always a uniform distribution $U[0,1]$

where, the joint CDF F is defined as the probability of the random variable X_i taking values less than or equal to x_i i.e;

$$F(x_1,x_2,...x_d) \overset{\text{def}}{=} P(X_1 \le x_1, X_2 \le x_2, ...X_d \le x_d) \tag{2}$$

In the above equations, x_i is a specific realization of the random variable X_i. Using the universal CDF property (Figure 2b), that the output of any continuous CDF is a uniform distribution, Equation 1 is equated to the standard copula notation. Here, similar to the random variable X_i, u_i represents the realizations of a uniform distribution $U[0,1]$.

If f is the multivariate probability density function (PDF) of the CDF F and f_i, the corresponding univariate PDFs of the CDFs F_i, then in terms of probability density functions Equation 1 can be written as follows:

$$f(x_1,x_2...x_d) = c(F_1(x_1),F_2(x_2),...F_d(x_d)) \prod_{i=1}^{d} f_i(x_i) \tag{3}$$

where,

$$c(u_1,...u_d) = \frac{\partial C(u_1,...u_d)}{\partial u_1 ... \partial u_d} \tag{4}$$

Therefore, from Equations 1 and 3 we can say that to represent any multivariate probability density function we need the following two sets of information: **(a)** *the univariate CDFs F_i of all the variables*, and **(b)** *corresponding Copula function $C(u_1,...u_i)$*.

Copula-based multivariate distribution modeling techniques generally approximate the function $C(.)$ using standard copula functions [53]. The most common among all the available copulas is the *Gaussian copula* function, which is derived from the *standard multivariate normal distribution*. For the purpose of data reduction in scientific datasets, Gaussian copula is well-suited because it requires storing only the correlation matrix of the data, which can be efficiently computed in an *in situ* environment.

Gaussian Copula: To set in terms of the above explanations, if F is a standard normal distribution of d-dimensions, then the corresponding $C(.)$ in equation 1 is a Gaussian copula. For a d-dimensional standard normal distribution $\mathcal{N}_d(\mathbf{0},\rho)$, with zero mean vector $\mathbf{0}$ and correlation matrix ρ the corresponding Gaussian copula function C^G_ρ with the parameter ρ can be denoted as;

$$C^G_\rho(u_1,...u_d) = \Phi_\rho(\Phi^{-1}(u_1),...\Phi^{-1}(u_d)) \tag{5}$$

where, Φ^{-1} represents the inverse CDF of a standard normal distribution and Φ_ρ represents the CDF of a multivariate standard normal distribution with correlation matrix ρ. Standard normal distributions have well-known closed-forms for the CDF functions, therefore, we can easily compute the Gaussian copula function using equation 5, provided we know the correlation matrix ρ. The final multivariate distribution, thus obtained, is often termed as *meta-Gaussian* distribution since the dependency structure is Gaussian but the marginals can be arbitrary distributions.

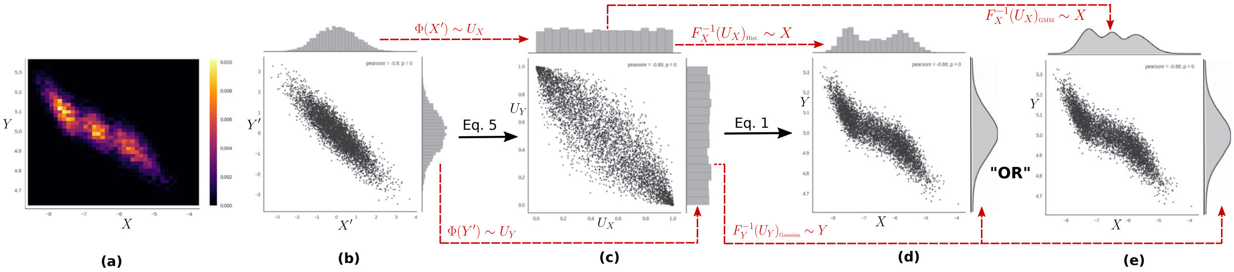


Fig. 3: Copula-based sampling example: (a) Joint distribution of the original bivariate samples with correlation coefficient -0.9. (b) Step 1: Generate new bivariate samples from a bivariate standard normal distribution. (c) Step 2: Construct the Gaussian copula with uniform marginals. (d,e) Step 3: Final bivariate samples with arbitrary univariate distribution types. A histogram representation for Y in (d) and a GMM representation for Y in (e), while, X is being modeled by a Gaussian distribution in both the scenario.

To summarize, the multivariate distribution-based data modeling stage of our proposed framework involves storing the desired *univariate distributions for the individual variables* and their *Gaussian copula parameters (i.e, ρ)* for each spatial partition in the simulation domain. Since, our objective is to reduce storage footprint, instead of storing the complete correlation matrix, ρ, which is a symmetric matrix, we store only the pairwise correlation coefficient of all the variables, which constitutes the lower and the upper triangles in the matrix. Therefore, for multivariate data with n variables, the overall storage of our proposed data summarization for a single partition can be written as;

$$S = \sum_{i=1}^{n} m_i + \binom{n}{2} \tag{6}$$

where, m_i is the storage footprint of the univariate distribution chosen for the i-th variable, while $\binom{n}{2}$ is the cost of storing the Gaussian copula parameter. We can optimally choose univariate distribution models for individual variables depending on factors like storage footprint (i.e., m_i) and computation times and estimate them in parallel.

Spatial Distributions: By storing only the value distributions of the physical variables in the simulation, we cannot retain the spatial context in the data. Spatial information is a vital property of scientific datasets and many analysis and visualization tasks require spatial queries and context of the data. Therefore, in our work, besides considering the physical variables, we also consider the spatial variables (i.e., x, y and z - dimensions) as part of our multivariate system. In other words, the effective number of variables in our system is $n = n_p + n_s$, where n_p is the number of physical variables computed in the simulation and n_s is the number of spatial variables (3 for a three-dimensional spatial model). We store the spatial variables in the form of *spatial distributions*. A benefit of using our copula-based flexible framework for storing the spatial distributions is that, for a regular partitioning, which is a popular partitioning scheme, we can use uniform distributions to model the spatial variables. Since copula functions have uniform marginals implicitly, we do not have to effectively store any extra information for the spatial distributions apart from their correlation coefficients with all the other variables. In the next section, we demonstrate the advantage of persevering spatial information for effective post-hoc analysis.

5 Post-hoc Multivariate Analysis and Visualization

In the second stage of our framework, to facilitate various post-hoc multivariate analyses using our constructed multivariate data summaries, we propose *multivariate sampling-based visualization* and *multivariate query-driven analysis* strategies. The key to performing such analysis in a flexible and scalable manner is to have an efficient copula-based sampling strategy. Therefore, we first explain in detail, with an simple bivariate example, the steps involved in sampling from a Gaussian copula-based multivariate model.

Copula-based Sampling Strategy: Consider a multivariate sample of two random variables X and Y, with a strong negative correlation ($\rho = -0.9$). The original joint distribution of the two variables is shown in Figure 3(a). Let F_X and F_Y be the CDFs of the desired univariate distribution respectively. As mentioned in the previous section, these univariate distributions can be of any arbitrary type (Histogram, GMM or Gaussian). Given F_X, F_Y and ρ(=-0.9), the three steps involved in our sampling method are as follows:

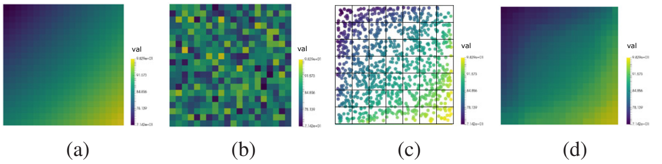


Fig. 4: Advantage of spatial distributions: (a) Original scalar field of resolution 20×20. (b) Scalar field resampled from a histogram, without any spatial information. (c) Samples generated by the copula-based strategy with spatial distributions. (d) Density field constructed from the copula-based samples in (c).

- **Step 1:** Generate new multivariate samples from a *standard bivariate normal distribution* with the correlation matrix ρ. Figure 3(b) shows the scatter plot view of the generated samples. In this step, the samples only preserve their correlation, while the univariate marginals are standard normal distributions with mean value 0 and standard deviation of 1.
- **Step 2:** The output of a CDF always follow a uniform distribution as illustrated in Figure 2(b). Using this property, we transform the bivariate samples generated in Step 1 to a bivariate uniform distribution as shown in Figure 3(c). By equation 5, these transformed samples, generated from a bivariate standard normal distribution represent the corresponding bivariate Gaussian copula. The dependency structure between the variables is still preserved but the marginals are uniform distributions.
- **Step 3:** Finally, we transform the uniform distributions of the two variables to the desired distribution types using the inverse functions of the precomputed CDFs F_X and F_Y. If we know the inverse CDF of a distribution, we can always transform uniform samples to the corresponding distribution, a fact, illustrated by Figure 2(a). As shown in Figure 3(d,e), the final bivariate samples (with sample $\rho = -0.88$) closely represent the initial bivariate samples. Since the transformation in this step takes place from uniform marginals, we can use arbitrary target univariate distribution for transformation. For example, F_X can be a Histogram (Figure 3(d)) or a GMM (Figure 3(e)), while F_Y is a Gaussian in the two alternatives.

For a d-dimensional multivariate system, we start Step 1 above with a d-dimensional standard normal distribution. Using this 3-step sampling strategy, we are able to generate multivariate samples from our proposed multivariate data summaries that preserve the correlation among the variables, an important property desired in any multivariate analysis task.

Advantage of Spatial Distributions: The multivariate samples generated from our proposed data summaries can be denoted as

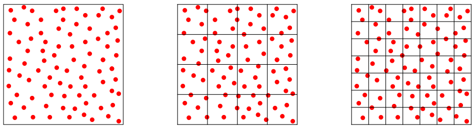

(a) samples (b) 5 × 5 (c) 8 × 8

Fig. 5: Arbitrary grid resolutions for the sample scalar fields.

$(v_1, ..., v_{n_p}, x, y, z)$, where, v_i's are the sample values for the n_p physical variables and (x, y, z), the corresponding sample location in the spatial domain. The spatial information associated with every sample not only facilitates post-hoc analysis but also strengthens dependency modeling accuracy of the copula functions. Figure 4 shows the results of a simple experiment to highlight the advantage of storing spatial distributions along with the value distributions of the physical variables. Consider, a small two-dimensional scalar field of resolution 20 × 20, with values linearly increasing along the diagonal from the top-left to the bottom-right corner of the field, as shown in Figure 4(a). Let, H_V be the histogram of the scalar value (say variable V). By sampling H_V, we get possible values of V, but without any spatial context. Therefore, if we visualize the generated random samples we get a noisy scalar field with similar value distribution, but inaccurate spatial information as shown in Figure 4(b). On the other hand, if we consider this as a three-dimensional multivariate system with variables V, X and Y, where X and Y are the spatial variables in the field, we are able to retain the spatial information in our generated samples (Figure 4c). Figure 4(d) shows the density field for the generated particle samples, where we are able to generate more accurate statistical realizations of the initial field. Moreover, since it is a regular Cartesian grid we can use uniform distribution to model X and Y.

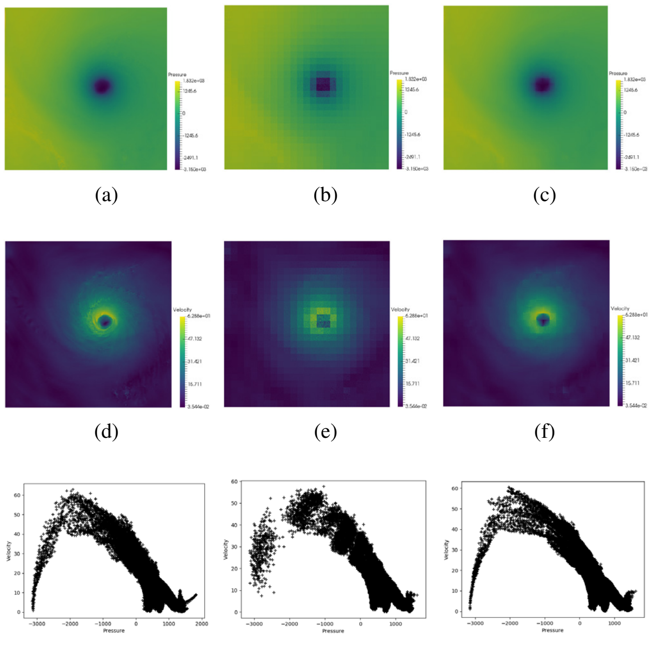
5.1 Multivariate Sampling-based Visualization

Visualizing the scalar fields of the individual variables in the form of volumes or surfaces is a common practice among scientists while dealing with multivariate data. In order to facilitate such visualizations using our proposed multivariate data summaries, we generate statistical realizations/samples from our data representation to create multivariate scalar fields that can be visualized as a replacement of the raw data. We generate multivariate samples for each partition in the spatial domain using our copula-based sampling strategy. Since the generated multivariate samples contain spatial locations, we can create the sample scalar fields by performing particle density estimation at the grid points [42]. For each multivariate sample, we assigned the distance-weighted average of the physical variables to the nearest grid point. The generated sample scalar fields can be in any arbitrary user-specified grid resolutions as illustrated in Figure 5. As a result, depending on the computational resources available on the analysis machine, users can specify a high or a low-resolution sample grid to visualize.

Algorithm 1 Generating a sample scalar field

```
1: $\mathscr{D} \leftarrow [D_1, ...D_p]$    ▷ list of distributions for $p$ partitions
2: $S_j[T_x, T_y, T_z] \leftarrow \mathbf{0}$    ▷ sample scalar field of size $(T_x, T_y, Tz)$
3: $sumOfWeights[T_x, T_y, T_z] \leftarrow \mathbf{0}$
4: for all $D_i$ in $\mathscr{D}$ do
5:     $\mathscr{S}[N] \leftarrow generateMVsamples(D_i, N)$    ▷ sample size $N$
6:     for all s in $\mathscr{S}[.]$ do    ▷ $\mathbf{s} \sim (s_1, .., s_n, s_x, s_y, s_z)$
7:         $(g_x, g_y, g_z) \leftarrow nearestGridLocation(s_x, s_y, s_z)$
8:         $dis \leftarrow distance(\{g_x, g_y, g_z\}, \{s_x, s_y, s_z\})$
9:         $weight \leftarrow 1/dis$
10:        $S_j[g_x, g_y, g_z] \mathrel{+}= (s_j * weight)$
11:        $sumOfWeights[g_x, g_y, g_z] \mathrel{+}= weight$
12: $S_j[.] \mathrel{/}= sumOfWeights[.]$    ▷ the final sample scalar field
```

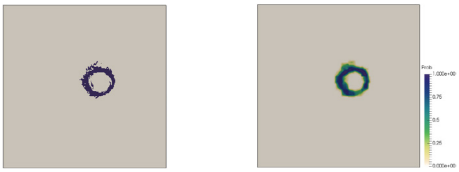
The pseudo-code in Algorithm 1 shows the steps involved in generating a sample scalar field. We create a sample scalar field S_j of user-specified target resolution (T_x, T_y, T_z) for the j^{th} variable in a system with n variables. For each multivariate data summary D_i (corresponding to each partition), we generate N multivariate samples using our copula-based sampling strategy as explain above, via the function $generateMVsamples(.)$ in line 5 of Algorithm 1. We then compute the distance-weighted average of the sample values of the physical variables (here s_j) to eventually create the final statistical realization of the scalar field, i.e, S_j. The number of samples generated, N, depends on the size of each partition and is generally kept higher than the number of grid points in the partition to get reliable results.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig. 6: Two dimensional slices (250 × 250) of Isabel dataset, partitioned into 10 × 10 blocks: (a) Original Pressure field. (b) Pressure field sampled from multivariate histogram. (c) Pressure field sampled using copula-based strategy. (d) Original Velocity field. (e) Velocity field sampled from multivariate histogram. (f) Velocity field sampled using copula-based strategy. (g) Scatter-plot view of original field. (h) Scatter-plot view of the fields sampled from multivariate histogram. (i) Scatter-plot view of the field sampled using copula-based strategy.

Using a simple two-dimensional real-world multivariate data, we demonstrate the effectiveness of our proposed method. We consider 2D slices (resolution 250 × 250) of Pressure and Velocity variables from the Hurricane Isabel dataset. The full volumetric datasets with 11 physical variables will be used later for extensive evaluation in Section 6. The original Pressure and Velocity scalar fields are shown in Figure 6(a) and (d) respectively, while Figure 6(g) shows the scatter-plot view of how the two variables are related. As can be seen, there is a non-linear relationship between the two variables. However, partitioning the spatial domain into smaller blocks help break down the complex global multivariate relationship into relatively simpler local relationships [38, 52], which can be accurately modeled by the Gaussian copula. In this example, we partition the spatial domain into regular blocks of size 10 × 10. To compare our copula-based strategy with a standard multivariate distribution based strategy, we compute multivariate histograms for the two variables Pressure and Velocity across all the partitions. Using our proposed framework, we only compute the univariate distributions of Pressure, Velocity and the two spatial dimensions X and Y. We use univariate histograms for Pressure and Velocity (with similar bin counts as the multivariate histogram, i.e., 64), while uniform distributions for X and Y. Also, we store the 6, i.e., $\binom{4}{2}$ correlation coefficients to capture the correlation matrix (parameter for Gaussian copula function). Figure 6(b) and (e) show the results of the sample scalar fields generated



(a)

(b)

Fig. 7: Multivariate Query-Driven Analysis: (a) The deterministic results of the query $-2000 < Pressure < 500$ and $40 < Velocity < 50$ in the original raw data. (b) Probabilistic result generated by our methods, i.e., $P(-2000 < Pressure < 500 \text{ AND } 40 < Velocity < 50)$.

with the multivariate histograms, while Figure 6(c) and (f) show the results from our copula-based sampling. The sample scalar fields are in the same resolution as the initial raw slices (250×250). Figure 6(h) and (i) show the corresponding scatter-plot views for the two cases. As can be seen, the copula-based sample scalar fields are able to closely resemble the complex multivariate relationship between Pressure and Velocity compared to just using a standard multivariate histogram. Therefore, the flexibility of adding the spatial information as extra variables in our multivariate model helps us to not only create a more accurate scalar field for the individual variables but also reliably capture their multivariate relationships.

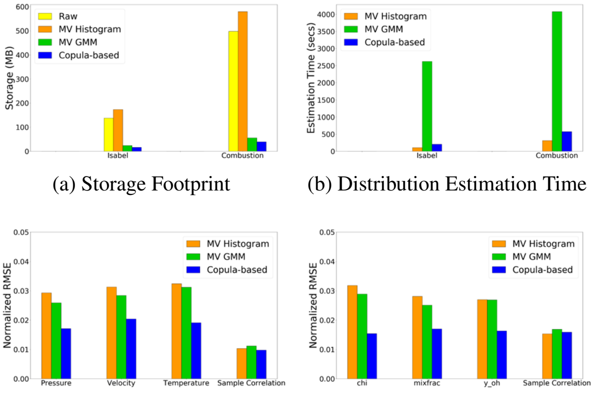
5.2 Multivariate Query-Driven Analysis

Query-driven analysis methods are a class of highly effective discovery visualization strategies [51]. They reduce the computational workload and the cognitive stress in large-scale scientific data by selecting regions of interest and filtering out the other non-pertinent regions. By focusing analysis and visualization efforts only on the regions of interest, such query-driven techniques make the work-flow of scientists more manageable and effective. For example, if scientists are interested in only a certain value range for two variables, a query-driven method helps them to focus only on the parts of the data that specifically meet their multivariate query, instead of looking at the entire simulation domain. They can further drill down into analyzing how the other variables behave in the region of interest to gain more insights. Many query-driven strategies rely on computing local data statistics to perform efficient query search operations [7, 24]. Therefore, the use of statistical data summaries is a wise choice for data reduction in large-scale simulations because it can easily facilitate such query-driven strategies. In this section, we explain in detail the process of performing multivariate query-driven analysis using our proposed multivariate data summaries.

To illustrate our copula-based multivariate query-driven analysis, consider the same 2D slices of the Isabel data used in Section 5.1. Consider performing a query on the Pressure range of $[-2000Pa - 500Pa]$ and Velocity range of $[40ms^{-1} - 50ms^{-1}]$. To compute the probability of seeing a multivariate value in this queried range, we selectively sample the stored multivariate distributions using our copula-based sampling method. To expedite the process, for each partition, we first check whether the corresponding univariate distributions of the queried variables satisfy the individual query ranges or not. We generate multivariate samples using our copula-based strategy only for the partitions which satisfy this initial check. As mentioned in the previous section, the multivariate samples generated in our method retains the spatial context in the form of spatial locations for each sample. By creating a spatial density field of the generated samples satisfying the query, we can produce the *probabilistic multivariate query field*, which highlights the probability of the specified multivariate query (i.e., $P(-2000 < Pressure < 500 \text{ AND } 40 < Velocity < 50)$). Figure 7(a) shows the region which satisfies the query in the original raw data. Figure 7(b) shows the corresponding probability density field for the query with probability values ranging from 0 to 1. A high value indicates a high possibility of seeing co-occurring Pressure and Velocity values in the specified ranges.

Table 1: Distribution Storage and Estimation Time

Dataset (Resolution)	#variables	Raw Size (MB)	block size	MV Histogram		MV GMM		Hybrid + Copula	
				Size (MB)	Est. Time (s)	Size (MB)	Est. Time (s)	Size (MB)	Est. Time (s)
Isabel (250x250x50)	11	137.5	5x5x5	173.1	106.1	23.7	2623.6	16.2	203.9
			7x7x7	152.5	111.5	8.13	4671.6	5.8	205.4
			10x10x10	113.7	98.2	2.95	5006.2	2.2	230.2
Combustion (480x720x120)	3	497.7	5x5x5	579.4	311.7	55.7	4077.7	39.2	573.3
			7x7x7	509.1	322.4	39.7	5150.4	14.3	561.7
			10x10x10	434.2	305.7	27.8	9708.5	5.1	583.6



(a) Storage Footprint

(b) Distribution Estimation Time

(c) RMSE for Isabel

(d) RMSE for Combustion

Fig. 8: Quantitative evaluation results for block size of 5^3.

6 Quantitative and Visual Evaluation

To demonstrate the effectiveness of our proposed multivariate data summarization strategy, we first evaluated it on two off-line multivariate data before applying it on a full-scale *in situ* simulation. We used the following off-line datasets: (a) Hurricane Isabel WRF model data of resolution $250 \times 250 \times 50$, with 11 physical variables, which models the development of a strong hurricane in the West Atlantic region, and (b) Combustion data of resolution $480 \times 720 \times 120$, with 3 physical variables, modeling a turbulent combustion process. For the purpose of our evaluation, we considered a single time step of the above datasets (time step 20 for Isabel and time step 30 for Combustion). All evaluations were performed on a standard workstation PC (Intel i7 at 3.40GHz and 16GB RAM).

Experiment Setup: In our experiment, we used non-overlapping regular partitioning scheme of equal block sizes to partition the simulation domain. Multivariate data summaries were then created for individual partitions. We tested our proposed summarization model against standard multivariate distribution models like multivariate histogram (sparse representation with 32 bins of equal width for all the dimensions) and multivariate GMM of 3 modes (with full covariance matrix). In our proposed flexible framework, to model the individual variables, we used a hybrid combination of univariate distributions involving *GMMs*, *Gaussian distributions* and *uniform distributions*, while, *Gaussian copula* was used to model the dependency among these hybrid distributions. For each partition, we performed a normality test (D'Agostino's K-squared test [13]) on the individual variables. For variables with a high certainty of following a normal distribution, we used a Gaussian distribution, else GMM of 3 modes was used, whereas, uniform distributions were used to model the spatial variables (i.e., x, y and z dimensions) for each partition. Therefore, the effective number of variables in our method for Isabel dataset is *14 (11 physical + 3 spatial)* and for the Combustion dataset is *6 (3 physical + 3 spatial)*.

Storage Footprint: The storage size of our proposed multivariate data summaries was significantly less as compared to the standard multivariate distributions, even when including the 3 spatial variables and extra indexing information for recording the hybrid univariate distribution types at each partition. Figure 8(a) compares the storage sizes for the three different models in the Isabel and Combustion datasets for block sizes of 5^3. Clearly, multivariate histogram is not a good alternative for the purpose of data-reduction. Also, the fact that in

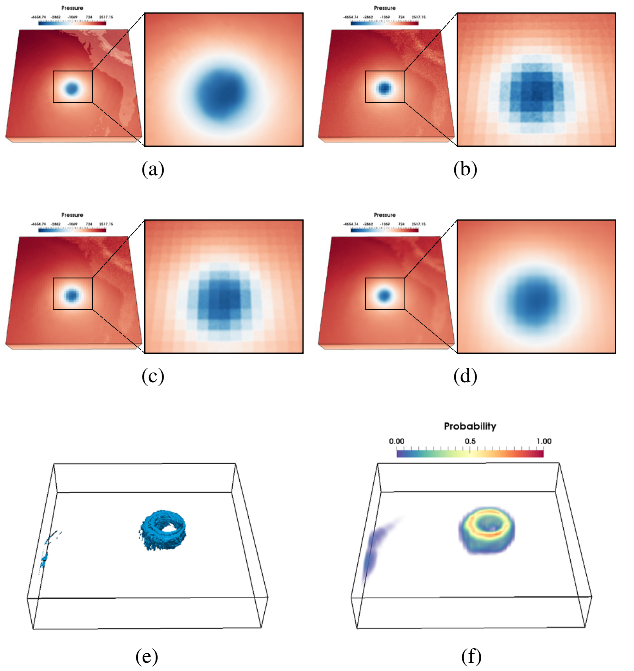


Fig. 9: Results from Isabel dataset for block size 5^3: (a) Original Pressure scalar field. (b) Pressure field constructed from multivariate histograms representation. (c) Pressure field constructed from multivariate GMM of 3 modes. (d) Pressure field created by our copula-based model, which retains the spatial context in the multivariate samples. (e) Region in the original raw data corresponding to the multivariate query of $-2000 < Pressure < 500$ and $40 < Velocity < 50$. (f) The probability field generated by our copula-based strategy for the similar query, i.e., $P(-2000 < Pressure < 500 \text{ AND } 40 < Velocity < 50)$.

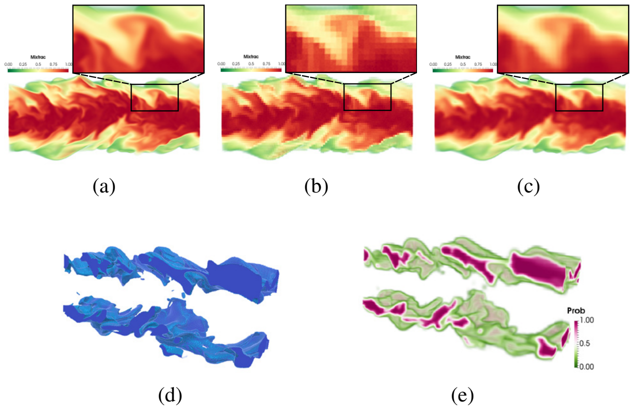


Fig. 10: Results from Combustion dataset for block size 5^3: (a) Original mixfrac scalar field. (b) Mixfrac field constructed from multivariate GMM of 3 modes. (c) Mixfrac field created by our copula-based model. (e) Region in the original raw data corresponding to the multivariate query of $0.3 < Mixfrac < 0.7$ and $y\_oh > 0.0006$. (f) The probability field generated by our copula-based strategy for the similar query, i.e., $P(0.3 < Mixfrac < 0.7 \text{ AND } y\_oh > 0.0006)$.

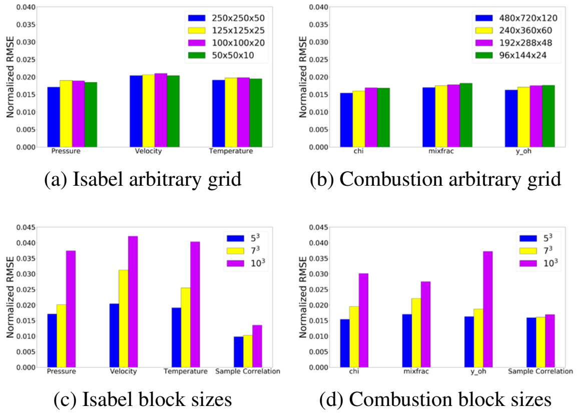


Fig. 11: (a) and (b) show the consistent RMSE values for different grid resolutions of the sample scalar field, when block size is 5^3. (c) and (d) show the trend of increasing RMSE values with increasing block-sizes.

our hybrid model, we selectively used GMMs of 3 modes and single Gaussian distributions, helps us achieve better storage size than the standard multivariate GMM (of 3 modes).

Estimation Time: We compared the estimation times of the three data summarization models for the two datasets. As shown in Figure 8(b), the distribution estimation time for multivariate GMM is significantly high compared to the other models. As a result, despite having good storage advantages, multivariate GMMs will greatly increase the simulation time when used in *in situ* applications. On the other hand, estimating multiple univariate distributions is comparatively less expensive, because of which our proposed multivariate data modeling strategy performed significantly better. The estimation time of our model included the time for normality test, the individual univariate distribution estimation and the Gaussian copula parameter computation time. Table 1 reports the storage sizes and estimation times for different block sizes.

Accuracy: Using the three data summarization models, we created sample scalar fields of resolutions similar to the original raw data. In the case of multivariate histogram and multivariate GMMs, for each grid location in the reconstructed field, we draw random samples from the distribution corresponding to the block (partition) that the grid location belongs to. The value of this sample is assigned to the specific grid location. This approach is similar to the reconstruction strategies employed in other univariate distribution-based data summarizations works [17, 58]. On the other hand, we employed the copula-based strategy explained in Section 5.1 to generate the sample scalar fields using our proposed data summaries. To compare the accuracies of the sample scalar fields, we computed their normalized root mean squared error (RMSE) with the corresponding original raw fields. Figure 8(c) and (d) show the RMSE results for three variables in both the datasets. The results of all the 11 variables for Isabel is provided in the supplementary material. To evaluate the multivariate relationship preserved by the models, we computed the RMSE values of the sample correlation coefficients of all the pairs of variables with the original correlation coefficients across all the partitions. As shown in the last stack of bar-charts in Figure 8(c) and (d), the sample correlation errors from the three different models are mostly similar, this is because, the use of copula is just another way of modeling multivariate distributions. Figure 9(a-d) show the visual comparison of the sample scalar field generated for the Pressure variable in Isabel dataset, while Figure 10(a-c) show the results for the Mixfrac variable in Combustion dataset (more results are provided in the supplementary material). The accuracy of scalar fields generated by our copula-based sampling strategy is better than the standard models because we were able to retain the spatial information in the form of spatial distributions. Therefore, based on the above three criteria, i.e., *storage footprint*, *estimation time* and *accuracy*, we can say that our proposed flexible multivariate data summary framework is better suited for the analysis of large-scale multivariate data than the corresponding standard multivariate distributions.

Multivariate Query: To facilitate query-driven analysis tasks, we computed the probability field for a given multivariate query using our hybrid model. Figure 9(e) shows the deterministic query result on the original raw data for the multivariate query $-2000 < Pressure < 500$ and $40 < Velocity < 50$ for the Isabel dataset. Figure 9(f) shows the corresponding probability field generated for the same query using our mul-

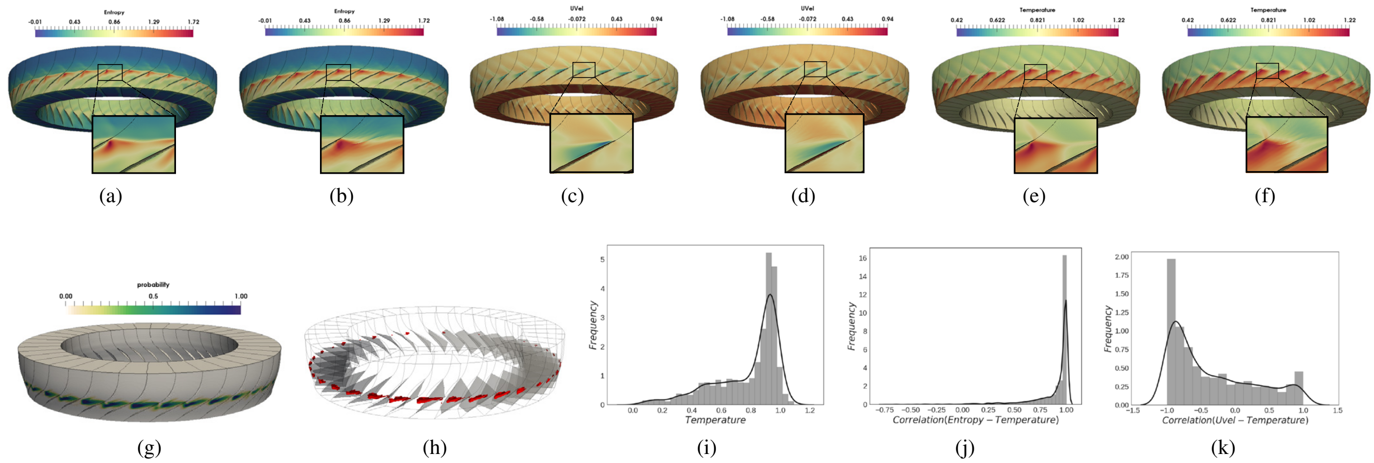


Fig. 12: Post-hoc analysis of the jet turbine dataset. (a) Original Entropy field. (b) Sample scalar field of Entropy. (c) Original Uvelocity field. (d) Sample scalar field of Uvelocity. (e) Original Temperature field. (f) Sample scalar field of Temperature. (g) Probabilistic multivariate query result i.e., $P(Entropy > 0.8 \text{ AND } Uvel < -0.05)$ (h) Isosurface for probability value 0.5. (i) Distribution of Temperature values in the queried region i.e., $P(Temp|Entropy > 0.8 \text{ AND } Uvel < -0.05)$. (j) Distribution of correlation coefficients between Entropy and Temperature for the queried region. (k) Distribution of correlation coefficients between Uvelocity and Temperature for the queried region.

tivariate data summaries (i.e., $P(-2000 < Pressure < 500 \text{ AND } 40 < Velocity < 50)$). As a result of the spatial information preserved in our model, we were able to successfully identify the region of interest for the specific query along with uncertainty information, provided in the form of the probability values. The regions with high probability value have higher chances of satisfying the given query. Based on the query results, scientists can further analyze the properties of other variables in this spatial range (more results are provided in the supplementary material). Similarly, Figure 10(d) shows the deterministic query results on the original Combustion raw data for the query $0.3 < mixfrac < 0.7$ and $y\_oh > 0.0006$, while, Figure 10(e) shows the corresponding probabilistic query ($P(0.3 < mixfrac < 0.7 \text{ AND } y\_oh > 0.0006)$).

Arbitrary Grid Resolution: The sample scalar fields generated by our method can be created in arbitrary user-specified grid resolutions because of the spatial information retained in the multivariate samples. As a result, users have the flexibility to create a high or a low-resolution sample field directly from the summaries depending on the computational resources available at their disposal for analysis. To test the results of the arbitrary grid resolutions, we computed the RMSE scores of the generated sample scalar fields with that of the corresponding scalar fields sub-sampled from the original raw field. Figure 11(a) shows the normalized RMSE scores for three variables in the Isabel dataset. For a single variable, each bar corresponds to the RMSE score of the corresponding grid resolution. The sub-sampled scalar field generated from the original raw data is considered as the baseline for each resolution size. Similarly, Figure 11(b) shows the results for Combustion dataset. The RMSE scores remain consistent across different grid resolutions for the individual variables.

Effect of block sizes: We also studied the effect of partition block sizes (i.e, granularity of domain partitioning) on the overall storage size, estimation time and RMSE values. With larger block sizes, the overall storage footprint decreases but the overall estimation time increases. This increase of estimation time is more significant with multivariate GMMs. Table 1 shows the storage and estimation times for different block sizes for the two test datasets. Also, with larger block sizes the overall RMSE values for the analysis results increases. Figure 11(c) and (d) show the increasing trend of RMSE values for some of the individual variables and the sample correlation coefficients in Isabel and Combustion datasets respectively. The number of multivariate samples generated from each multivariate data summary also depends on the partition block size. To get statistically reliable results the number of samples is generally larger than the number of grid points in each partition. We tested with different sample sizes and observed that with increasing sample sizes the overall accuracy does not differ significantly after a certain size. For our case, we used sample sizes of 500, 1000 and 1500 for block sizes of 5^3, 7^3 and 10^3 respectively.

7 In situ Application and Domain Expert Feedback

Based on the positive evaluation results in off-line multivariate data, next, we applied our proposed flexible multivariate data summarization framework on a real-world *in situ* environment. Using our proposed model, we want to facilitate flexible and scalable multivariate analysis of data generated in a large-scale computational fluid dynamics (CFD) simulation code, TURBO [9, 10]. TURBO, developed at NASA, is a Navier-Stokes based, time-accurate CFD simulation code to study transonic jet engine compressors at high resolutions. Domain experts compute various physical variables to study and analyze the inception of flow instability across the compressor blades. Flow instability can lead to potential stalls in the engine, which can damage the blades. Therefore, it is important to understand and analyze what roles the different variables play in the creation of such unstable flow structures. However, the computational cost and the amount of data produced from a single simulation is quite significant, which makes such multivariate analysis very unwieldy and overwhelming for the scientists.

For this case, scientists were interested in analyzing the multivariate relationship among the variables Entropy, Uvelocity and Temperature. We computed our proposed multivariate data summaries for partitions of size 5^3 across the simulation domain. Based on the results of normality test, we used either a Gaussian distribution or a GMM (with 3 modes) to model the univariate distribution of individual variables. The spatial variables were modeled using uniform distributions, while Gaussian copula captured the dependency structure among all these variables (i.e., 6, 3 physical + 3 spatial). The *in situ* simulation was performed in a cluster (Oakley [5], at the Ohio Supercomputer Center) containing 694 nodes with Intel Xeon x5650 CPUs (12 cores per node), and 48 GB of memory per node. The simulation was run on 328 cores in total. We executed 2 full revolutions of the jet turbine, resulting in 7200 time steps. *In situ* multivariate data summarization was performed every 10^{th} time step, thereby storing 720 time steps. We created our hybrid multivariate data summaries by accessing the simulation memory directly without additional data copies. The domain of the compressor consists of 36 blade passages, each with a spatial resolution of $151 \times 71 \times 56$. The simulation outputs raw data in multiblock PLOT3d format of size 690 MB per time step, which accounts for **496.8** GB for just two 2 revolutions. On the other hand, our proposed multivariate data summaries result in only **19.6** GB of total storage footprint. Table 2 shows the overall simulation times for our *in situ* application. Our multivariate data summary creation process requires

Table 2: *In situ* Performance

Simulation Time (hrs)	Raw I/O Time (hrs)	In situ Data Summarization (hrs)	Data Summaries I/O Time (hrs)
13.5	1.76	2.09	0.0063

Table 3: Post-hoc analysis performance

MV Query per time step(secs)	Sample Scalar Field per time step (secs)	Normalized RMSE		
		Entropy	U-Vel	Temp
64.6	178.3	0.0211	0.0174	0.0184

about 15.4% of the original simulation time but offers the flexibility of scalable post-hoc analysis as compared to storing the raw data (the raw data I/O time itself takes 13% of the simulation time).

Multivariate data summaries were later used to generate sample scalar fields for the variables of interest, as well as perform multivariate query-driven analysis. Figure 12(a,c,e) show the original scalar fields for Entropy, Uvelocity and Temperature respectively, whereas Figure 12(b,d,f) shows the corresponding sample scalar fields for the respective variables generated by our copula-based sampling strategy. Scientists were interested to see how the selected variables affect flow instability in the turbine. Prior studies on univariate data [8, 14, 17] highlights that Entropy values great than 0.8 and negative Uvelocities correspond to potentially unstable flow structures. Therefore, we computed the multivariate query, $Entropy > 0.8$ and $Uvel < -0.05$ from our stored data summaries. The corresponding probability field is shown in Figure 12(g), whereas, Figure 12(h) shows the isosurfaces of probability value 0.5 across the blade structures. Figure 12(i) shows the distribution of Temperature values in this queried region (i.e., $P(Temp|Entropy > 0.8 \text{ AND } Uvel < -0.05)$). The peak in the distribution suggests that Temperature values around 0.9 can be related to potential flow instability. Figure 12(j) and (k) show how Temperature is correlated with Entropy and Uvelocity respectively, in the selected queried range. There is a strong positive correlation with Entropy and a substantial amount of negative correlation with Uvelocity. Such exploratory analysis activity can help the scientists to gain more insights into the multivariate relationships in their simulation. All post-hoc analysis were performed on a standard workstation PC (Intel i7 at 3.40GHz and 16GB RAM) with 8 CPU cores. Using OpenMP parallelization, we ran the analysis tasks on all the CPU cores. Table 3 shows the average post-hoc analysis time and accuracy results for a single time step.

Domain Expert Feedback: We presented the results and explained the idea behind of our proposed framework to the domain scientist. The expert agrees with the fact that having a summarized version of the original multivariate data is useful, as it facilitates effective post-hoc multivariate analysis. Previous analysis works on this simulation were primarily centered around studying the effect of the variables independently [8,14,17], but our expert feels that this framework will be useful to study how the interaction among different variables influence flow instability in the engine. The result of our multivariate query aligns with the expert's knowledge that the potential unstable regions generate near the edges of the blades, as shown in Figure 12(g,h). The expert feels that the distribution of Temperature and correlation strengths in this queried region is similar to what is originally expected. Generally, because of the large storage requirements, the raw simulation data was stored only after around 25-30 time steps. But, with our proposed data summaries, we can now store at finer temporal resolutions (every 10^{th} time step in this case). Expert feels that this will help analyze the finer temporal events in the simulation. Overall, the expert acknowledges that our proposed framework is an effective strategy to understand the multivariate relationships in his simulation without having to store the large-scale simulation data off-line.

8 Discussion

In this section, we would like to briefly discuss upon and highlight some of the important aspects of our work.

Evaluation: The two major contributions of our work are (a) the use of copula functions to model multivariate distributions and (b) the use of spatial distributions in a scalable manner. In Section 6, we evaluated the effects of both of these contributions by testing against standard multivariate distribution models. There are a large number of distribution-based data summarization approaches for univariate data [14, 17, 56, 58]. However, to the best of our knowledge, not much work has been done to address large-scale multivariate data. Therefore, to evaluate our proposed framework, we compared it with the standard multivariate distribution models like multivariate histograms and multivariate GMMs. On one hand, as a result of the copula functions, we are able to get both *better data reduction rates* as well as *better estimation times* as compared to the standard models. On the other hand, as a result of incorporating the spatial distributions, without extra overhead, we are able to generate *more accurate and statistically reliable* results during the post-hoc analysis phase. An important point to note here is that, by including the spatial attributes in the standard multivariate histograms and multivariate GMMs, we can expect to get similar post-hoc analysis results as ours, but the rapid increase in storage size (for multivariate histograms) and estimation times (for multivariate GMMs) with addition of new dimensions make such an approach infeasible in practical context of *in situ* applications.

Modeling Individual Variables: With respect to modeling the individual dimensions (variables), we can use any type/family of distribution as long as it has a well-defined continuous CDF. In our work, when generating the multivariate samples, we generated sample values for all the variables and created their individual sample scalar fields, but, if needed, the scientists can also independently pick just the univariate distributions of one variables to study. As a result, it offers the flexibility to perform other state-of-the-art distribution-based analysis [15,45,58] for univariate data as well.

Modeling Dependency Structure: The dependency among the variables are modeled using Gaussian copula, which essentially involves computing the correlation matrix for the variables. Gaussian copula functions model only the linear relationships among the variables, however, for this work, where we partition the simulation domain into smaller block sizes this does not possess serious limitations. As was shown in the example in Section 5.1, we were able to closely model the overall non-linear relationship between the Pressure and Velocity fields of the Isabel dataset. There are other standard copula functions which can capture special types of multivariate relationships [53]. For example, Clayton copula can capture a heavy left-tail dependency among the variables, Gumbel copula captures right-tail dependencies, Student-t copula can simultaneously capture both tail dependency structures. However, different copula functions have different parameter requirements. In our framework, the Gaussian copula function can be replaced with any other copula function depending on the kind of relationship that exists between the variables. We plan to incorporate such studies in our future endeavors. For this work, where storage footprint and estimation time are crucial requirements of the framework, Gaussian copula is the most cost-efficient and effective alternative.

9 Conclusion and Future Work

In this paper, we have proposed a flexible copula-based distribution-driven analysis framework for large-scale multivariate data. The proposed framework offers an effective solution to summarize multivariate data *in situ*. In future, we plan to investigate other distribution-driven problems in the field of scientific visualization that can benefit from the flexibility of different kinds of available copula functions. Another important research problem is to facilitate multivariate distribution-based feature tracking, where the users specify a feature distribution to look for in the data. Modeling and analyzing multivariate ensemble data and uncertain vector fields using copula-based strategies are also in our plan of activities in future.

Acknowledgments

This work was supported in part by US Department of Energy Los Alamos National Laboratory contract 47145, UT-Battelle LLC contract 4000159447, NSF grants IIS- 1250752, IIS-1065025, and US Department of Energy grants DE- SC0007444, DE- DC0012495, program manager Lucy Nowell.

References

[1] J. Ahrens, S. Jourdain, P. O'Leary, J. Patchett, D. H. Rogers, and M. Petersen. An image-based approach to extreme scale in situ visualization and analysis. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '14, pages 424–434, Piscataway, NJ, USA, 2014. IEEE Press.

[2] T. Athawale, E. Sakhaee, and A. Entezari. Isosurface visualization of data with nonparametric models for uncertainty. *IEEE Trans. Vis. Comput. Graph.*, 22(1):777–786, 2016.

[3] A. C. Bauer, H. Abbasi, J. Ahrens, H. Childs, B. Geveci, S. Klasky, K. Moreland, P. O'Leary, V. Vishwanath, B. Whitlock, and E. W. Bethel. In situ methods, infrastructures, and applications on high performance computing platforms. *Computer Graphics Forum*, 35(3):577–597, 2016.

[4] W. Bethel, L. Gosink, K. Joy, and J. Anderson. Variable interactions in query-driven visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13:1400–1407, 09 2007.

[5] O. S. Center. Oakley supercomputer. `http://osc.edu/ark:/19495/hpc0cvqn`, 2012.

[6] J. Chanussot, A. Clement, B. Vigouroux, and J. Chabod. Lossless compact histogram representation for multi-component images: application to histogram equalization. In *IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No.03CH37477)*, volume 6, pages 3940–3942 vol.6, July 2003.

[7] A. Chaudhuri, T. H. Wei, T. Y. Lee, H. W. Shen, and T. Peterka. Efficient range distribution query for visualizing scientific data. In *2014 IEEE Pacific Visualization Symposium*, pages 201–208, March 2014.

[8] C. M. Chen, S. Dutta, X. Liu, G. Heinlein, H. W. Shen, and J. P. Chen. Visualization and analysis of rotating stall for transonic jet engine simulation. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):847–856, Jan 2016.

[9] J. Chen, R. Webster, M. Hathaway, G. Herrick, and G. Skoch. Numerical simulation of stall and stall control in axial and radial compressors. In *44th AIAA Aerospace Sciences Meeting and Exhibit*. American Institute of Aeronautics and Astronautics, 2006.

[10] J.-P. Chen, M. D. Hathaway, and G. P. Herrick. Prestall behavior of a transonic axial compressor stage via time-accurate numerical simulation. *Journal of Turbomachinery*, 130(4):041014, 2008.

[11] U. Cherubini and E. Luciano. Bivariate option pricing with copulas. *Applied Mathematical Finance*, 9:69–85, 2002.

[12] H. Childs. Data exploration at the exascale. *Supercomputing frontiers and innovations*, 2(3), 2015.

[13] R. B. D'agostino, A. Belanger, and R. B. D. Jr. A suggestion for using powerful and informative tests of normality. *The American Statistician*, 44(4):316–321, 1990.

[14] S. Dutta, C. M. Chen, G. Heinlein, H. W. Shen, and J. P. Chen. In situ distribution guided analysis and visualization of transonic jet engine simulations. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):811–820, Jan 2017.

[15] S. Dutta and H.-W. Shen. Distribution driven extraction and tracking of features for time-varying data analysis. *IEEE Trans. on Vis. and Comp. Graphics*, 22(1):837–846, 2016.

[16] S. Dutta, H. W. Shen, and J. P. Chen. In situ prediction driven feature analysis in jet engine simulations. In *2018 IEEE Pacific Visualization Symposium (PacificVis)*, pages 66–75, April 2018.

[17] S. Dutta, J. Woodring, H. W. Shen, J. P. Chen, and J. Ahrens. Homogeneity guided probabilistic data summaries for analysis and visualization of large-scale data sets. In *2017 IEEE Pacific Visualization Symposium (PacificVis)*, pages 111–120, April 2017.

[18] G. Elidan. *Copulas in Machine Learning*, pages 39–60. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[19] P. Embrechts, F. Lindskog, and A. McNeil. Modelling dependence with copulas and applications to risk management. *Handbook of heavy tailed distributions in finance*, 8(1):329–384, 2003.

[20] N. Fabian, K. Moreland, D. Thompson, A. C. Bauer, P. Marion, B. Gevecik, M. Rasquin, and K. E. Jansen. The paraview coprocessing library: A scalable, general purpose in situ visualization library. In *2011 IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, pages 89–96, 2011.

[21] F. Ferstl, K. Bürger, and R. Westermann. Streamline variability plots for characterizing the uncertainty in vector field ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):767–776, Jan 2016.

[22] R. Fuchs and H. Hauser. Visualization of multivariate scientific data. *Computer Graphics Forum*, 28(6):1670–1690.

[23] R. Fujimaki, Y. Sogawa, and S. Morinaga. Online heterogeneous mixture modeling with marginal and copula selection. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 645–653, New York, NY, USA, 2011. ACM.

[24] L. Gosink, C. Garth, J. Anderson, E. Bethel, and K. Joy. An application of multivariate statistical analysis for query-driven visualization. *IEEE Trans. on Vis. and Comp. Graphics*, 17(3):264–275, 2011.

[25] F. Han and H. Liu. Semiparametric principal component analysis. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 171–179. Curran Associates, Inc., 2012.

[26] F. Han and H. Liu. High dimensional semiparametric scale-invariant principal component analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(10):2016–2032, Oct 2014.

[27] S. Hazarika, A. Biswas, and H. W. Shen. Uncertainty visualization using copula-based analysis in mixed distribution models. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):934–943, Jan 2018.

[28] H. Jänicke, A. Wiebel, G. Scheuermann, and W. Kollmann. Multifield visualization using local statistical complexity. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1384–1391, Nov 2007.

[29] M. Jarema, I. Demir, J. Kehrer, and R. Westermann. Comparative visual analysis of vector field ensembles. In *2015 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 81–88, Oct 2015.

[30] D. Kao, A. Luo, J. L. Dungan, and A. Pang. Visualizing spatially varying distribution data. In *Proceedings of the Sixth International Conference on Information Visualisation, 2002*, pages 219–225, 2002.

[31] S. Kirshner and B. Póczos. Ica and isa using schweizer-wolff measure of dependence. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 464–471, New York, NY, USA, 2008. ACM.

[32] H. Lehmann and B. Jung. In-situ multi-resolution and temporal data compression for visual exploration of large-scale scientific simulations. In *IEEE 4th Symposium on Large Data Analysis and Visualization (LDAV), 2014*, pages 51–58, 2014.

[33] S. Liu, J. Levine, P. Bremer, and V. Pascucci. Gaussian mixture model based volume visualization. In *2012 IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, pages 73–77, 2012.

[34] K. Lu and H.-W. Shen. A compact multivariate histogram representation for query-driven visualization. In *Proceedings of the 2015 IEEE 5th Symposium on Large Data Analysis and Visualization (LDAV)*, LDAV '15, pages 49–56, 2015.

[35] C. Lundstrom, P. Ljung, and A. Ynnerman. Local histograms for design of transfer functions in direct volume rendering. *IEEE Trans. on Vis. and Comp. Graphics*, 12(6):1570–1579, 2006.

[36] A. Luo, D. Kao, and A. Pang. Visualizing spatial distribution data sets. In *Proceedings of the Symposium on Data Visualisation 2003*, VISSYM '03, pages 29–38, 2003.

[37] J. Ma and Z. Sun. Copula component analysis. *CoRR*, abs/cs/0703095, 2007.

[38] A. Mahalanobis, B. Vijaya, and A. Nevel. Volume correlation filters for recognizing patterns in 3d data, 2001.

[39] A. J. McNeil, R. Frey, and P. Embrechts. *Quantitative risk management: concepts, techniques and tools*. Princeton series in finance. Princeton University Press, Princeton (N.J.), 2005.

[40] R. B. Nelsen, J. J. Quesada-Molina, J. A. Rodriguez-Lallena, and M. Úbeda-Flores. On the construction of copulas and quasi-copulas with given diagonal sections. *Insurance: Mathematics and Economics*, 42:473–483, 2008.

[41] M. Otto and H. Theisel. Vortex analysis in uncertain vector fields. In *Computer Graphics Forum*, volume 31, pages 1035–1044. Blackwell Publishing Ltd, 2012.

[42] T. Peterka, H. Croubois, N. Li, E. Rangel, and F. Cappello. Self-adaptive density estimation of particle data. *SIAM Journal on Scientific Computing*, 38(5):S646–S666, 2016.

[43] T. Pfaffelmoser, M. Reitinger, and R. Westermann. Visualizing the positional and geometrical variability of isosurfaces in uncertain scalar fields. In *Computer Graphics Forum*, volume 30, pages 951–960. Wiley Online Library, 2011.

[44] K. Pöthkow and H. C. Hege. Positional uncertainty of isocontours: Condition analysis and probabilistic measures. *IEEE Transactions on Visualiza-*

tion and Computer Graphics, 17(10):1393–1406, Oct 2011.

[45] K. Pöthkow and H.-C. Hege. Nonparametric models for uncertainty visualization. *Computer Graphics Forum*, 32(3pt2):131–140, 2013.

[46] K. Pöthkow, C. Petz, and H.-C. Hege. Approximate level-crossing probabilities for interactive visualization of uncertain isocontours. *International Journal for Uncertainty Quantification*, 3(2), 2013.

[47] K. Potter, J. Kniss, R. Riesenfeld, and C. R. Johnson. Visualizing summary statistics and uncertainty. *Computer Graphics Forum (Proceedings of Eurovis 2010)*, 29(3):823–831, 2010.

[48] K. Potter, J. Krüger, and C. Johnson. Towards the visualization of multidimensional stochastic distribution data. In *Proceedings of The International Conference on Computer Graphics and Visualization (IADIS) 2008*, 2008.

[49] M. Rey. Copula models in machine learning. 2015.

[50] M. Rey and V. Roth. Copula Mixture Model for Dependency-seeking Clustering. *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 927–934, 2012.

[51] O. Ruebel, E. W. Bethel, M. Prabhat, and K. Wu. Query-driven visualization and analysis. 2012.

[52] N. Sauber, H. Theisel, and H. p. Seidel. Multifield-graphs: An approach to visualizing correlations in multifield scalar data. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):917–924, Sept 2006.

[53] T. Schmidt. Coping with Copulas. *Copulas - From Theory to Applications in Finance*, (15):1–23, 2006.

[54] A. Sklar. *Fonctions de repartition a n dimensions et leurs marges*. 1959.

[55] A. Tewari, M. J. Giering, and A. Raghunathan. Parametric characterization of multimodal distributions with non-gaussian modes. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 286–292, Dec 2011.

[56] D. Thompson, J. A. Levine, J. C. Bennett, P. T. Bremer, A. Gyulassy, V. Pascucci, and P. P. Pbay. Analysis of large-scale scalar data using hixels. In *Large Data Analysis and Visualization (LDAV), 2011 IEEE Symposium on*, pages 23–30, 2011.

[57] V. Vishwanath, M. Hereld, and M. E. Papka. Toward simulation-time data analysis and i/o acceleration on leadership-class systems. In *2011 IEEE Symposium on Large Data Analysis and Visualization (LDAV)*, pages 9–14, 2011.

[58] K. C. Wang, K. Lu, T. H. Wei, N. Shareef, and H. W. Shen. Statistical visualization and analysis of large data using a value-based spatial distribution. In *2017 IEEE Pacific Visualization Symposium (PacificVis)*, pages 161–170, April 2017.

[59] B. Whitlock, J. M. Favre, and J. S. Meredith. Parallel in situ coupling of simulation with a fully featured visualization system. In *Proceedings of the 11th Eurographics Conference on Parallel Graphics and Visualization*, EGPGV '11, pages 101–109. Eurographics Association, 2011.

[60] P. C. Wong and R. D. Bergeron. 30 years of multidimensional multivariate visualization. In *Scientific Visualization, Overviews, Methodologies, and Techniques*, pages 3–33, Washington, DC, USA, 1997. IEEE Computer Society.

[61] J. Woodring, J. Ahrens, J. Figg, J. Wendelberger, S. Habib, and K. Heitmann. In-situ sampling of a large-scale particle simulation for interactive visualization and analysis. In *Proceedings of the 13th Eurographics / IEEE - VGTC Conference on Visualization*, pages 1151–1160. Eurographics Association, 2011.

[62] J. Woodring, M. Petersen, A. Schmeißer, J. Patchett, J. Ahrens, and H. Hagen. In situ eddy analysis in a high-resolution ocean climate model. *IEEE Trans. on Vis. and Comp. Graphics*, 22(1):857–866, 2016.

[63] Y. C. Ye, T. Neuroth, F. Sauer, K. L. Ma, G. Borghesi, A. Konduri, H. Kolla, and J. Chen. In situ generated probability distribution functions for interactive post hoc visualization and analysis. In *2016 IEEE 6th Symposium on Large Data Analysis and Visualization (LDAV)*, pages 65–74, Oct 2016.

[64] H. Yu, C. Wang, R. W. Grout, J. H. Chen, and K. L. Ma. In situ visualization for large-scale combustion simulations. *IEEE Computer Graphics and Applications*, 30(3):45–57, 2010.