Uncertainty-aware Prediction in Spatio-temporal Simulation Ensemble Visualizations

Marina Evers and Lars Linsen
Westfälische Wilhelms-Universität Münster, Germany

Keywords: Uncertainty Visualization, Ensemble Visualization, Parameter-Space Visualization, Volume Visualization.

Abstract: Spatio-temporal simulation ensembles are used to investigate the dependence of the simulation behavior on input parameters. Running simulations for a large number of input parameter settings is computationally expensive. We propose a scheme for exploring the parameter space using predictions of simulation outcomes and estimating the uncertainty in the predictions. The prediction approximates the simulation result by interpolating feature vectors of existing runs. The feature vectors are used to compute similarities between simulation runs facilitating visualization of the entire ensemble within a 2D (or 1D-over-time) multi-dimensional scaling embedding. Uncertainties of the prediction are computed based on distance, interpolation and diversity, which are visually encoded by an uncertainty band in the embedding. To guide the user to choose suitable parameter settings for prediction, we also propose a parameter-space visualization of the uncertainty. The approach is applied to real-world data simulating deep-water impact of asteroids.

1 INTRODUCTION

Spatio-temporal ensemble visualizations are common in many areas of science like physics or geography. These simulations typically depend on initial conditions or input parameters. To understand the dependence of the simulation result on the initial parameters, it is commonly necessary to create an ensemble including various simulation runs.

Visualizing an ensemble comes with the challenge of handling large amounts of data and establishing a visual encoding that allows for the comparative analysis of simulation runs. Another challenge inherent to ensembles is to determine, which parameter settings to choose for desirable simulation outcomes. We propose an approach that facilitates the analysis of the parameter space of spatio-temporal simulation ensembles by predicting simulation outcomes of new parameter settings from existing ones. The method is computationally inexpensive as it is based on the interpolation of feature vectors, see Section 4. The predicted result is visualized within the embedding proposed by Fofonov et al. (Fofonov et al., 2016), see Section 3.

It is important to visually convey how good we expect the prediction to be. Thus, we compute the uncertainty of the prediction based on multiple factors such as distance to existing settings in the parameter space, the uncertainty introduced by the interpolation scheme, and the diversity of the existing results used for interpolation, see Section 5. We visualize the uncertainty in the form of an uncertainty band in the embedding. Moreover, we visualize the uncertainty of possible predictions in the parameter space, which facilitates choosing suitable parameter settings for prediction (and eventually for running the actual simulation) in the sense of computational steering, see Section 6.

We apply our approach within a real-world scenario of investigating the influence of multiple input parameters on the result of an ensemble of deep water impact by asteroid simulations, see Section 7.

2 RELATED WORK

Many visualization approaches for spatio-temporal simulation ensembles display statistical information such as mean or variance of the ensemble (Potter et al., 2009) (Sanyal et al., 2010), which does not support the comparative analysis of individual simulation runs nor the analysis of the influence of simulation parameters. Phadke et al. (Phadke et al., 2012) proposed techniques to explore and compare ensemble members, but they are limited to small numbers. Fofonov et al. (Fofonov et al., 2016) introduced the concept

Evers, M. and Linsen, L.
Uncertainty-aware Prediction in Spatio-temporal Simulation Ensemble Visualizations.
DOI: 10.5220/0007344702160224
In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2019)*, pages 216-224
ISBN: 978-989-758-354-4

of *multi-run plots* that allows for visualizing simultaneously the temporal evolution of all simulation runs within an ensemble. We will build upon this idea, see Section 3. Applications of the multi-run plot include the visual representation of ensembles to detect patterns and outliers (Fofonov and Linsen, 2018a). Our goal is to predict the outcome of new simulation runs and determine their uncertainty. Fofonov et al. neither provide information beyond the scope of the simulated run nor do they support choosing parameters for further simulation runs. Different tools for computational steering exist in the analysis of ensemble data such as World Lines (Waser et al., 2010) and ComVis (Matkovic et al., 2008). Splechtna et al. (Splechtna et al., 2015) introduced an interactive visual analysis with hierarchical steering that supports multi-level simulation models. However, these approaches are only applicable if the computation of a simulation run does not take too much time.

To obtain a better understanding of the relationship between the parameter space and the target space in an interactive setting, Berger et al. (Berger et al., 2011) proposed a method using statistical learning to predict further results and apply it to car engine design. In the following years, Sedlmair et al. (Sedlmair et al., 2014) provided an abstract conceptual framework. Potter et al. (Potter et al., 2017) proposed an ensemble steering framework similar to the conceptional framework that uses approximations of simulations based on machine learning to explore, design, and plan energy systems. They also mention the possibility of a later simulation to eventually produce a more accurate result. None of the mentioned approaches for parameter space visualization deals with multi-run spatio-temporal simulation ensembles that are common for physical simulations, which is what we are addressing in this paper.

Recently, there has been active research on the visualization of uncertainty, cf. (Pang et al., 1997) (Griethe and Schumann, 2006) (Potter et al., 2012) (Liu et al., 2017)(Rhodes et al., 2003)(Xie et al., 2015). Some of them are within the context of simulation ensembles such as Noodles (Sanyal et al., 2010) or EnsembleVis (Potter et al., 2009) for analyzing the associated uncertainty in the data. Both tools assign the uncertainty by computing the standard deviation (or variance) to the means using statistically pre-processed data. We, instead, propose novel approaches for estimating uncertainty in the parameter space and respective visualizations and for prediction outcomes in the embedded space.

3 BACKGROUND

Multi-run Plot. Our approach is based on the visual representation of the simulation ensemble within an embedding as proposed by Fofonov et al. (Fofonov et al., 2016). Each simulation run is depicted as a curve parametrized over time, where positions on the curve reflect similarities of the time steps of the runs. To compute the multi-run plots, Fofonov et al. (Fofonov et al., 2016) propose to compute the similarity of two scalar fields by computing the Jaccard distance of the volumes that are enclosed by the isosurfaces of a representative isovalue. They later generalize this isosurface similarity measure to a field similarity measure, which does not require the choice of a representative isovalue, and even to a multifield similarity measure that simultaneously considers multiple fields (Fofonov and Linsen, 2018b). We support all options for similarity measures.

In case of the isosurface similarity, the values at the samples are thresholded against the isovalue leading to a binary n-dimensional vector, where n is the number of samples. In the case of field similarity, the field values are stored in the n-dimensional vector. We refer to this vector as *feature vector* and will use it for our predictions, see Section 4.

The multi-run plot is generated by using pairwise dissimilarities of all time steps of all simulation runs, which are stored in a distance matrix. The distance matrix is fed to an MDS approach (Wickelmaier, 2003). By drawing a curve through the points in the 2D embedding that represent the time steps of each simulation run in chronological order, one obtains the multi-run plot, where each run is typically colored using a distinct color. An alternative to generating a 2D embedding is to generate a 1D embedding such that an axis orthogonal to the 1D embedding can be used to show the change over time in a 2D plot. Figure 1 shows such 1D-over-time multi-run plots, while Figure 2(left) shows a 2D multi-run plot.

Interpolation. Our prediction is based on scattered data interpolation schemes. Obviously, the choice of the interpolation method affects the outcome. We choose representatives of the most common scattered data interpolation schemes based on inverse distances, radial basis functions, and natural neighbors.

The original inverse-distance-weighted interpolation method was proposed by Shepard (Shepard, 1968). Calculating a weighted average of the values at n given sample points $\boldsymbol{p}_i$, the interpolated function value $f(\boldsymbol{q})$ in point $\boldsymbol{q}$ can be written as

$$f(\boldsymbol{q}) = \begin{cases} \frac{\sum_{i=1}^{n}(d_i)^{-u} f(\boldsymbol{p}_i)}{\sum_{i=1}^{n}(d_i)^{-u}}, & \text{if } d_i \neq 0 \\ f(\boldsymbol{p}_i), & \text{if } d_i = 0 \end{cases} ,$$

where $d_i = \|\boldsymbol{q} - \boldsymbol{p}_i\|$ is the Euclidean distance to data point $\boldsymbol{p}_i$, $f(\boldsymbol{p}_i)$ is the value of data point $\boldsymbol{p}_i$, and u is an exponent. Shepard empirically chose $u = 2$.

The interpolation method using radial basis functions also uses weighted averages, but with radially symmetric basis functions as a weight. It calculates

$$f(\boldsymbol{q}) = \frac{\sum_{i=1}^{n} \phi(\|\boldsymbol{q} - \boldsymbol{p}_i\|) f(\boldsymbol{p}_i)}{\sum_{i=1}^{n} \phi(\|\boldsymbol{q} - \boldsymbol{p}_i\|)},$$

where $\phi : \mathbb{R}_+ \to \mathbb{R}$ is a basis function (Buhmann, 2000). We are using Gaussian kernels $\phi(d_i) = e^{-(\varepsilon d_i)^2}$ and multiquadrics $\phi(d_i) = \sqrt{1 - (\varepsilon d_i)^2}$ where d_i is defined as above and ε is a parameter.

The natural neighbor interpolation scheme (Sibson, 1980) is based on the construction of a Voronoi diagram. Points with adjacent Voronoi cells are referred to as natural neighbors. Sibson's interpolant f in point $\boldsymbol{q}$ is defined as

$$f(\boldsymbol{q}) = \frac{\sum_{i=1}^{k} u_i f(\boldsymbol{p}_i)}{\sum_{i=1}^{k} u_i},$$

where k is the number of natural neighbors and weight u_i is defined as the ratio between the area of the intersection of the Voronoi cell of $\boldsymbol{p}_i$ with the Voronoi cell of $\boldsymbol{q}$ when inserting $\boldsymbol{q}$ in the Voronoi diagram (Park et al., 2006) (Cueto et al., 2003).

4 PREDICTION

Our goal is an interactive visual analysis of the parameter space of a spatio-temporal simulation ensemble including the influence of parameters on the simulation outcome. The parameter space is typically sampled using some heuristic sampling strategy. During the analysis, it is desirable to create new simulation runs by querying the parameter space, i.e., by selecting a new sample interactively. Ideally, the simulation would then be run to produce a new result. However, in practice, this is not possible due to computation times. We propose to predict the simulation outcome by interpolating between the existing simulation results. Thus, the user can select a new parameter setting interactively, our prediction is executed for this parameter setting, and the predicted result is visualized in the multi-run plot (cf. Section 3).

The predicted outcome is obtained by predicting each time step individually. Each time step is interpolated from the respective time steps of all existing simulation runs. However, different simulation runs may have different simulated duration and adaption of time steps may be used leading to varying lengths of time steps. To obtain a consistent prediction, the start time of a prediction is chosen as the largest available start time among the existing runs and the end time corresponds to the smallest end time. Linear interpolation between time steps was chosen to obtain the function values for time steps that lie between two known time steps. The time steps on which the interpolated results are calculated are chosen equidistantly.

For the interpolation, we perform a scattered data interpolation, where the weights are computed using Euclidean distance in parameter space. We employ the different interpolation schemes described in Section 3, of which the user can select any.

Since the predicted result is visualized in the multi-run plot, we propose to only interpolate the feature vectors (cf. Section 3). The feature vectors of all existing simulation runs are pre-computed and stored, i.e., during our interactive session only the feature vectors need to be loaded and not the entire fields. The predicted feature vectors are used to compute the dissimilarities for the MDS process. By predicting each time step we can directly produce a visualization of the predicted result in the form of a curve in the multi-run plot. Hence, we can interactively analyze the parameter space using our interpolation scheme of feature vectors for the multi-run plot.

The feature vector interpolation is a compromise between interpolating the entire field and interpolating just the embedded 2D points, which represents a trade-off between accuracy (spatial features are still represented well) and efficiency (only n-dimensional vectors need to be loaded and processed).

5 UNCERTAINTY ESTIMATION

Since our predictions only approximate a simulation result, it is assumed to introduce errors. We do not know a priori, how large these errors are, but we can determine uncertainties to estimate how accurate we expect our predictions to be. Assuming that we want to predict the result for parameter values out of a d-dimensional parameter space, then the selected parameter settings are represented by a point $\boldsymbol{q}$ in the d-dimensional parameter space. We identified three sources of uncertainty that we consider to estimate the uncertainty of a prediction for parameter setting $\boldsymbol{q}$. First, the higher the distance of parameter setting $\boldsymbol{q}$ to parameter settings of existing runs, the more uncertain the prediction gets. Second, the higher the diversity of existing simulation runs with similar parameter setting to $\boldsymbol{q}$, the more uncertain the prediction gets. Third, the choice of the interpolation method itself potentially creates uncertainty.

Distance Uncertainty. When predicting the result for an existing parameter setting using interpolation, the prediction will be equal to the existing result. Hence, we can be sure about the outcome and the uncertainty should be zero. The farther we go away from the existing parameter settings, the more the uncertainty should grow. Moreover, the uncertainty should vary smoothly and its values should range from zero to one.

We can capture this distance-based uncertainty by computing the Euclidean distances d_i of the predicted parameter setting $\boldsymbol{q}$ to the parameter settings $\boldsymbol{p}_i$ of the N existing runs used for interpolation. In the visualisation, the distance-based uncertainty is additionally multiplied by a scaling constant, see Section 6. We define our uncertainty measure by multiplying these distances and normalizing them by the largest possible distance d_{max}, i.e., we set

$$u_D = \prod_{i=1}^{N} \frac{d_i}{d_{max}}.$$

The multiplicative dependence on the distances assures that the uncertainty is zero, iff $\boldsymbol{q}$ is equal to any $\boldsymbol{p}_i$. Moreover, uncertainty increases smoothly with increasing distances.

Diversity Uncertainty. If all existing simulation runs for parameter settings close to the predicted parameter setting $\boldsymbol{q}$ are similar, then the interpolated prediction will also be similar and we can assume this to be a quite accurate prediction. The more the existing simulation runs for parameter settings close $\boldsymbol{q}$ vary, the more uncertain it is whether the prediction is producing an accurate variant. This uncertainty aspect we capture by computing the standard deviation of the feature vectors $f(\boldsymbol{p}_i)$ from the interpolated feature vector $f(\boldsymbol{q})$. The added terms are the squared differences of the feature vectors, which represent the squared lengths of the difference vectors. Additionally, each of the added terms is weighted with a function ϕ_i depending on the Euclidean distance d_i of $\boldsymbol{p}_i$ to $\boldsymbol{q}$ in parameter space such that parameter settings closer to $\boldsymbol{q}$ have a larger impact. We compute the diversity-based uncertainty measure as

$$u_V(\boldsymbol{q}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \phi_i (f(\boldsymbol{p}_i) - f(\boldsymbol{q}))^2}$$

where N is the number of points taken into consideration for the interpolation and

$$\phi_i = \frac{\frac{1}{d_i}}{\sum_k \frac{1}{d_k}} N \qquad \text{if all } d_k \neq 0.$$

This equation fulfills the condition $\sum_i \phi_i = N$, which assures that the result equals the standard deviation (without weights ϕ_i). If $d_j = 0$ for any j, ϕ_j is set to N and all other weights $\phi_k = 0$ for $k \neq j$. This assures that the uncertainty $u_V(\boldsymbol{q}) = 0$ when $d_j = 0$, i.e., $\boldsymbol{q} = \boldsymbol{p}_j$.

Interpolation Uncertainty. Interpolation is used for prediction. Every interpolation scheme makes an assumption that is reflected by the underlying interpolation model. Since we cannot be certain, which is the correct interpolation model, choosing one of them introduces uncertainty. To capture this uncertainty aspect, we compute the interpolation result by using other interpolation schemes and calculate the standard deviation of the other interpolation results to the chosen one, i.e.:

$$u_I(\boldsymbol{q}) = \sqrt{\frac{1}{N_I}\sum_{i=1}^{N_I} (f_i(\boldsymbol{q}) - f(\boldsymbol{q}))^2} \tag{1}$$

where N_I is the number of other interpolation methods used and $f_i(\boldsymbol{q})$ are the interpolated feature vectors using the other interpolation methods. If all interpolation methods produce the same results, i.e., $f_i(\boldsymbol{q}) = f(\boldsymbol{q})$ for all i, then the uncertainty $u_I(\boldsymbol{q}) = 0$ as desired.

Composited Uncertainty. For compositing the three uncertainty measures, we postulate that the overall uncertainty should be larger than zero, if any of its components is larger than zero. Thus, we composite the three uncertainty measures by

$$u(\boldsymbol{q}) = u_I(\boldsymbol{q}) + u_V(\boldsymbol{q}) + u_D(\boldsymbol{q}). \tag{2}$$

Please note that the uncertainty $u(\boldsymbol{q}) = 0$ when $\boldsymbol{q}$ coincides with the parameter setting of any existing run ($\boldsymbol{q} = \boldsymbol{p}_i$), as all three components vanish.

6 UNCERTAINTY VISUALIZATION

Prediction Uncertainty. Our first uncertainty visualization goal is to visually convey how certain a prediction is. Since we are visualizing the predicted result in the multi-run plot, the goal is to enhance the plot to visually represent the uncertainty in the prediction. Following the idea of a box plot and its generalizations in the form of functional box plots (Sun and Genton, 2011) or curve box plots (Mirzargar et al., 2014), we visualize the uncertainty in the prediction by displaying a band (e.g., reflecting the standard deviation) around the curve that represents the prediction (i.e., the most likely case).

For the creation of the band, we estimate the uncertainty of the predicted value for each time step.

The respective uncertainty value is used as the width of the band to both sides of the curve that represents the prediction. The band is colored using the same color that was used for the predicted curve, but with transparency to not occlude other curves.

To compute the band, we perform the computation of the uncertainties component-wise, i.e., for each component of the feature vector. The band is then created by adding the component-wise uncertainties to the interpolated feature vector and mapping the resulting vectors to the embedding of the multi-run plot. This mapping is performed for each time step independently. Connecting the mapped positions in chronological order leads to one border of the band, which is then mirrored to produce the band.

We want to emphasize that the components u_I and u_V are not normalized to a given range such that the results calculated for these components have the same dimension as the given data values. Therefore, they do not only provide a qualitative measure of uncertainty but also a quantitative one, i.e., it can directly be compared to the given data values. The uncertainty component u_D caused by the distance, however, is not directly correlated to the given data value ranges, which makes it primarily suitable for a qualitative analysis. In our experiments, we empirically chose a scaling factor for the values of u_D, i.e., using a division by factor 2^{14-n} with n being the number of simulation runs.

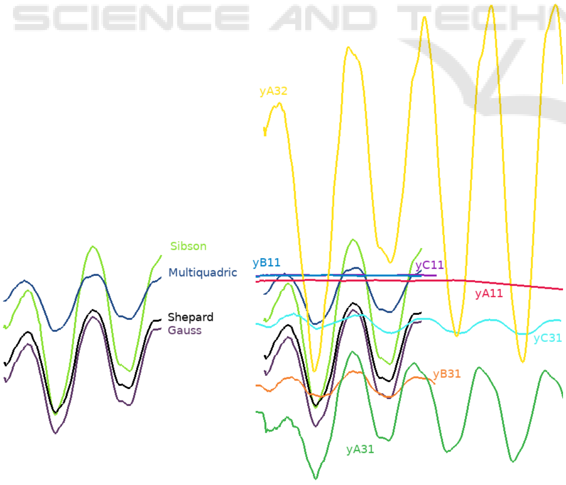

Figure 1: Comparison of interpolation methods for prediction of new run (height 1 km, radius of 240 m, angle of 50°) using all existing runs in the 1D-over-time multi-run plot. Predicted runs are qualitatively similar (left) and reflect the oscillatory behavior of existing runs (right).

Parameter Space Uncertainty. Since the user may select any parameter setting for prediction, guidance to select new parameter settings is desirable. This can be achieved by displaying the parameter settings of existing runs in the d-dimensional parameter space to see their distribution. To further guide the user to potentially interesting parameter settings, we visualize the uncertainty in the parameter space.

We first need to estimate the uncertainties. To do so, we sample the d-dimensional parameter space using an equidistant sampling strategy, which leads to a d-dimensional uncertainty field defined over a regular grid. For each grid point, we compute the uncertainty that is associated with the respective parameter setting using the uncertainty measures of Section 5. More precisely, we estimate the predicted simulation outcome and its uncertainty as in Section 6. As the uncertainty in Section 6 was computed per time step, we now average the uncertainty over all time steps to obtain a single scalar uncertainty value that is associated with each grid point of the parameter space sampling.

We propose to visualize the given field by interactively choosing an axis-aligned 2D slice through the parameter space and to perform a color mapping of the uncertainty values. We refer to this concept as the *uncertainty map*. A 2D slice-based approach is desirable, as it has no inherent occlusion issues and scales to dimensionality d beyond three dimensions. The resolution of the slice can be chosen by the user and trilinear interpolation is used to interpolate the uncertainty at any point on the slice.

We implemented our entire system within Voreen (Meyer-Spradow et al., 2009), a rapid application development framework for interactive volume visualization. All prediction and uncertainty visualization methods have been incorporated into the framework for a smooth user experience. Figure 6 shows the slice-based visualization of an uncertainty map.

7 APPLICATION SCENARIO

We apply our approach to the simulation ensemble of deep water impact by asteroids that hit the Earth's surface in the ocean (Patchett and Gisler, 2017), where each simulation run is represented by time-varying volumetric multi-field data with 300^3 spatial grid points and varying number of adaptive time steps (between 162 and 487). The ensemble data is made available through the IEEE SciVis Contest 2018, unfortunately only comprising seven runs with three scalar fields (pressure, temperature, and volume fraction of water). The parameter space consists of three parameters: the height of the airburst (zero in case of no airburst), the size of the asteroid, and the angle of entry. Table 1 lists the seven runs and their parameter settings. We used multifield similarities (Fofonov and Linsen, 2018b) such that all fields are considered and we did not have to choose representative isovalues.

Table 1: Parameter settings of Deep Water Impact simulation ensemble.

Run	Radius	Angle	Height
yA11	100m	45°	0km
yA31	250m	45°	0km
yA32	250m	60°	0km
yB11	100m	45°	5km
yB31	250m	45°	5km
yC11	100m	45°	10km
yC31	250m	45°	10km

Prediction. To document that the interpolation-based prediction is producing suitable results, we first made a simple test that makes prediction considering only two simulation runs, yA31 and yC31. The two runs have the same parameter setting for asteroid radius and incoming angle, but differ in the height of the airburst, where yA31 had no airburst and yC31 had an airburst at height 10km. We predicted runs that were in between by sampling the airburst height equidistantly with step size 2km. Figure 3 shows the prediction when using inverse-distance-based interpolation and 1D multi-run plots over time. For all our examples, the predicted line stops as soon as one of the existing runs used for interpolation has no further data. The results indeed deliver the expected and desired results, as the predicted runs smoothly vary between the existing runs when varying the simulation parameter.

Next, we wanted to investigate the behavior of the prediction when using different interpolation methods on all existing runs within the 3D parameter space. The parameter settings for the predicted run were an airburst at height 1 km, an asteroid radius of 240 m, and an entry angle of 50°. The 1D-over-time multi-run plots in Figure 1(left) show that all predictions are qualitatively similar by exhibiting an oscillatory behavior, but location and amplitude of the curves vary. Inverse-distance-based and Gaussian radial-basis-function interpolation produce the most similar results. When comparing the different prediction results to the existing runs in Figure 1(right), one can observe that the oscillatory behavior of the existing runs is preserved.

To test the quality of our prediction in comparison to running an actual simulation, we performed a leave-one-out test. Thus, we removed simulation run yC11 from the ensemble, then tried to predict run yC11, and compared the predicted to the simulated outcome. Note that the prediction is actually performing an extrapolation here. Still, all interpolation methods are applicable except for the natural neighbor interpolation, where our implementation required the predicted parameter setting to lie inside the convex hull of the existing parameter settings. Figure 2 shows the predicted results in comparison to the simulated result (labeled as yC11) for the water fraction field. The left side of the figure shows a 2D multi-run plot, where time in encoded by the brightness of the colors. We observe that the multiquadric radial-basis-function interpolation is producing results that are far from yC11, while Gaussian radial-basis-function interpolation and Shepard inverse-distance-based interpolation are very close. Since the variation of those runs is small when compared to other runs, we interactively select only the predicted runs and run yC11 and visualize them in a 1D-over-time multi-run plot on the right side of the figure. We observe that the Gaussian radial-basis-function interpolation produces a prediction that is very close to the simulated run, as the curves are almost coinciding. Similar observations can be made for the other fields such that we can conclude that the Gaussian radial-basis-function prediction is in surprisingly high agreement with the actual simulated run.

Prediction Uncertainty. To illustrate the prediction uncertainty visualization within the multi-run plots, we predict the pressure field of run yB31 using inverse-distance-based interpolation. Figure 4 shows the bands with respect to various uncertainties. The distance-based uncertainty shall be interpreted qualitatively only, i.e., it is more meaningful for comparing different predictions as in the uncertainty map (see below). The interpolation-based uncertainty visualization illustrates that there is some variation in the prediction when using different methods, which matches our observations from above. Still, the simulated run is not completely within the band such that we can conclude that the different interpolation methods generally do not predict the outcome as well as for run yC11 above. The diversity-based uncertainty shows up to be larger. This is caused by the fact that we only have six existing runs (when leaving out yB31), whose parameter settings have similar distances in parameter space. Since the existing runs are quite diverse in the simulation outcome, the diversity-based uncertainty is high throughout the prediction. Thus, the complete uncertainty obtained by compositing the three uncertainty aspects is also pretty high. The respective band (partially cropped) embeds run yB31.

Parameter Space Uncertainty. The uncertainty within the 3D parameter space is visualized using the slice-based views shown in Figure 6, where the X-axis represents the airburst height (0km-10km), the Y-axis represents the asteroid's radius (100m-250m), and the

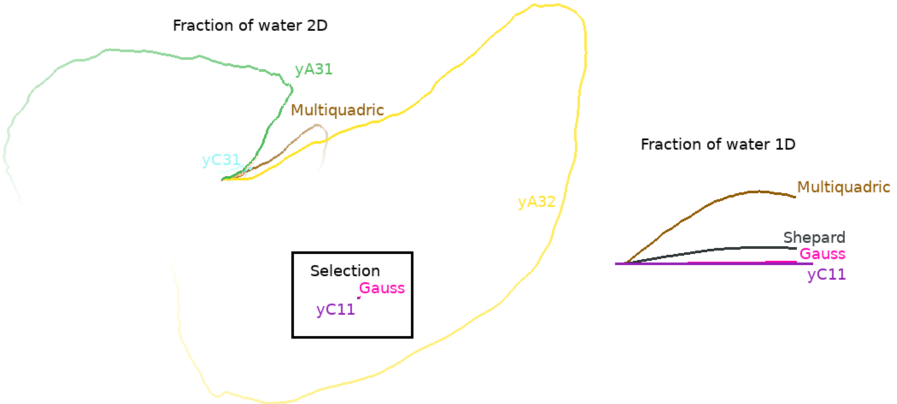

Figure 2: Interpolation results for the fraction of water using different methods for the parameters of the run yC11. In the left image, the time is colour coded. In the box with the black boundary, a selection of only the interpolation result produced with Gaussian kernels and the original run can be seen. It is located in the point where all the other runs also start.

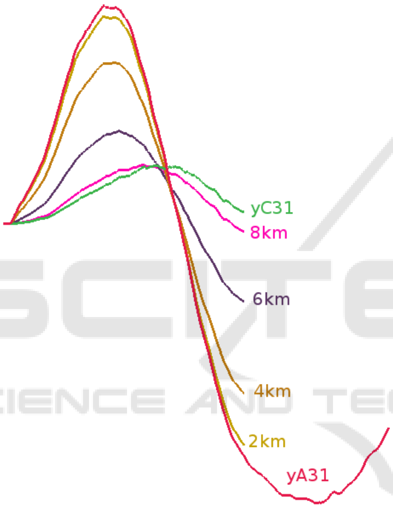

Figure 3: Prediction by interpolating between two runs (yA31 and yC31) delivers a smooth transition between the runs in the 1D-over-time multi-run plot. Inverse-distance-based interpolation with exponent 2 was chosen.

Z-axis represents the entry angle (45°-60°). The 3D uncertainty field within the parameters' range was estimated using the introduced uncertainty measures at 5^3 samples.

Figure 6a shows the distance-based uncertainty. It can be observed how the uncertainty increases with increasing distance from the existing runs' parameter settings depicted by the black dots (two black dots are occluded by the plane).

Figure 6b shows a different distribution of uncertainties when considering diversity. The largest uncertainty can be observed around the point with height 0 km, angle 60°, and radius 175 m. By construction, uncertainties are low close to parameter settings of existing runs, which can be observed. However, it can also be observed that uncertainties are quite high close to run yA32 (the only one with angle 60°) in the upper left corner of the slice. This was to be expected, as the simulation outcome of this run is quite different from the others.

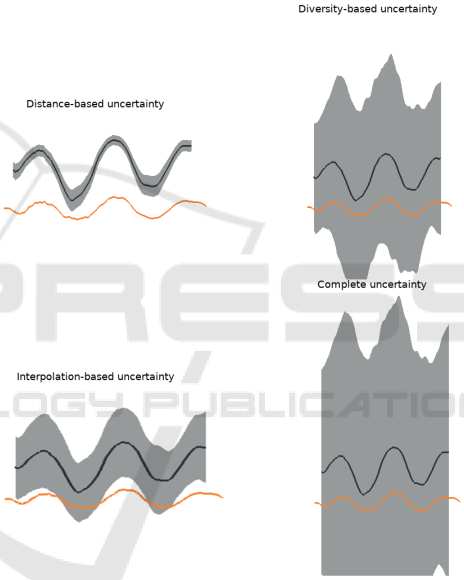

Figure 4: Uncertainty visualization in 1D-over-time multi-run plots for predicting the pressure field of run yB31 (orange). The prediction (black) is obtained using Shepard's inverse-distance-based interpolation.

This effect is even stronger for the interpolation-based uncertainty visualization in Figure 6c, where the highest uncertainties are computed in the larger vicinity of run yA32. The large differences between run yA32 and other runs lead to strongly different interpolation results. It is also interesting that the uncertainties for height 5km are less than for height 0km despite the fact that 3 runs with height 0km exist and only 2 runs with height 5km. Thus, the interpolation-based uncertainty is not mainly influenced by dis-

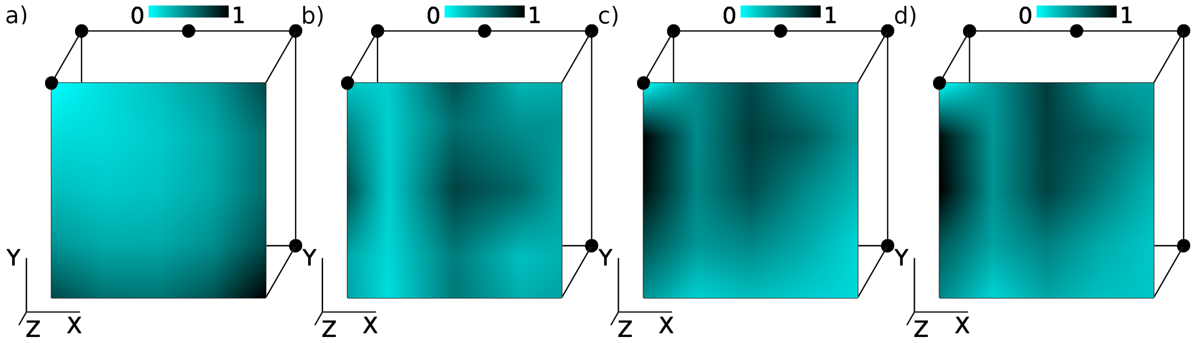

Figure 6: Visualization uncertainty in parameter space (X: airburst height, Y: asteroid radius, Z: entry angle) using 2D slices (Z=60°) through the uncertainty maps: a) Distance-based uncertainty. b) Diversity-based uncertainty. c) Interpolation-based uncertainty. d) Composited uncertainty.

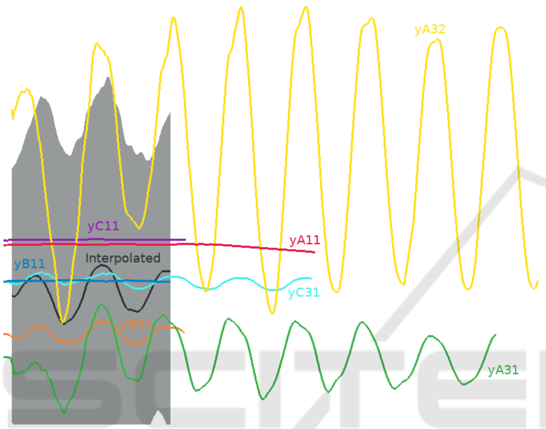

Figure 5: Composited uncertainty visualization in 1D-over-time multi-run plots for predicting the pressure field of run yB31 (orange). The prediction (black) is obtained using Shepard's inverse-distance-based interpolation.

tances.

The composited uncertainty map is shown in Figure 6d. It combines the different aspects of the uncertainty measures. We can conclude that the desired properties are achieved: Uncertainty vanishes at the parameter settings of existing points, it increases with increasing distance in their vicinity, and the uncertainty field is smooth.

8 DISCUSSION AND CONCLUSION

We proposed a method for predicting simulation outcomes by interpolating feature vectors and for estimating and visualizing the uncertainties in the prediction. The uncertainty map helps to identify good choices for further simulation runs in the sense of computational steering. In our experiments, the Gaussian radial-basis-function interpolation produced the best prediction results. However, one should also look into the nature of the analyzed physical simulations. The simulation outcome shall vary smoothly with changing parameter settings. In future work, it would be interesting to apply the approach to larger data sets. Computation times may then be a challenge again. Generating the uncertainty map over a high-resolution sample grid (especially when also parameter space is of higher dimension) would also require quite some computation time. Still, the computation times are expected to be several order of magnitudes lower than running the actual simulations. One way of speeding the computations up would be the usage of an adaptive time stepping scheme for prediction and uncertainty estimation.

ACKNOWLEDGMENTS

This work was partially supported by the Deutsche Forschungsgemeinschaft (DFG) under contract LI 1530/21-1.

REFERENCES

Berger, W., Piringer, H., Filzmoser, P., and Gröller, E. (2011). Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction. *Comput Graphics Forum*, 30(3):911–920.

Buhmann, M. D. (2000). Radial basis functions. *Acta Numerica*, 9:1–38.

Cueto, E., Sukumar, N., Calvo, B., Martínez, M., Cegonino, J., and Doblaré, M. (2003). Overview and recent advances in natural neighbour galerkin methods. *Archives of computational methods in engineering*, 10(4):307–384.

Fofonov, A. and Linsen, L. (2018a). Multivisa: Visual analysis of multi-run physical simulation data using interactive aggregated plots. In *Proceedings of the 13th International Joint Conference on Computer Vision,* *Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2018) - Volume 3: IVAPP, Funchal, Madeira, Portugal, January 27-29, 2018.*, pages 62–73.

Fofonov, A. and Linsen, L. (2018b). Projected field similarity for comparative visualization of multi-run multi-field time-varying spatial data. *Computer Graphics Forum*.

Fofonov, A., Molchanov, V., and Linsen, L. (2016). Visual analysis of multi-run spatio-temporal simulations using isocontour similarity for projected views. *IEEE Trans. Visual Comput. Graphics*, 22(8):2037–2050.

Griethe, H. and Schumann, H. (2006). The visualization of uncertain data: Methods and problems. *Proceedings of International Conference on Perspectives in Business Informatics Research*, 2006.

Liu, S., Maljovec, D., Wang, B., Bremer, P. T., and Pascucci, V. (2017). Visualizing high-dimensional data: Advances in the past decade. *IEEE Trans. Visual Comput. Graphics*, 23(3):1249–1268.

Matkovic, K., Gracanin, D., Jelovic, M., and Hauser, H. (2008). Interactive visual steering - rapid visual prototyping of a common rail injection system. *IEEE Trans. Visual Comput. Graphics*, 14(6):1699–1706.

Meyer-Spradow, J., Ropinski, T., Mensmann, J., and Hinrichs, K. (2009). Voreen: A rapid-prototyping environment for ray-casting-based volume visualizations. *IEEE Computer Graphics and Applications*, 29(6):6–13.

Mirzargar, M., Whitaker, R. T., and Kirby, R. M. (2014). Curve boxplot: Generalization of boxplot for ensembles of curves. *IEEE Transactions on Visualization & Computer Graphics*, 20(12):2654–2663.

Pang, A. T., Wittenbrink, C. M., and Lodha, S. K. (1997). Approaches to uncertainty visualization. *The Visual Computer*, 13(8):370–390.

Park, S. W., Linsen, L., Kreylos, O., Owens, J. D., and Hamann, B. (2006). Discrete sibson interpolation. *IEEE Transactions on visualization and computer graphics*, 12(2):243–253.

Patchett, J. M. and Gisler, G. R. (2017). Deep water impact ensemble data set. *http://sciviscontest2018.org/*.

Phadke, M. N., Pinto, L., Alabi, O., Harter, J., Taylor, R. M., Wu, X., Petersen, H., Bass, S. A., and Healey, C. G. (2012). Exploring ensemble visualization. *Proc.SPIE*, 8294:8294 – 8294 – 12.

Potter, K., Rosen, P., and Johnson, C. R. (2012). From quantification to visualization: A taxonomy of uncertainty visualization approaches. In Dienstfrey, A. M. and Boisvert, R. F., editors, *Uncertainty Quantification in Scientific Computing*, pages 226–249, Berlin, Heidelberg. Springer Berlin Heidelberg.

Potter, K., Wilson, A., Bremer, P. T., Williams, D., Doutriaux, C., Pascucci, V., and Johnson, C. R. (2009). Ensemble-vis: A framework for the statistical visualization of ensemble data. In *2009 IEEE International Conference on Data Mining Workshops*, pages 233–240.

Potter, K. C., Brunhart-Lupo, N. J., Bush, B. W., Gruchalla, K. M., Bugbee, B., and Krishnan, V. K. (2017). Coupling visualization, simulation, and deep learning for ensemble steering of complex energy models: Preprint.

Rhodes, P. J., Laramee, R. S., Bergeron, R. D., Sparr, T. M., et al. (2003). Uncertainty visualization methods in isosurface rendering. In *Eurographics*, volume 2003, pages 83–88.

Sanyal, J., Zhang, S., Dyer, J., Mercer, A., Amburn, P., and Moorhead, R. (2010). Noodles: A tool for visualization of numerical weather model ensemble uncertainty. *IEEE Trans. Visual Comput. Graphics*, 16(6):1421–1430.

Sedlmair, M., Heinzl, C., Bruckner, S., Piringer, H., and Möller, T. (2014). Visual parameter space analysis: A conceptual framework. *IEEE Trans Visual Comput Graphics*, 20(12):2161–2170.

Shepard, D. (1968). A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM National Conference*, ACM '68, pages 517–524, New York, NY, USA. ACM.

Sibson, R. (1980). A vector identity for the dirichlet tessellation. *Mathematical Proceedings of the Cambridge Philosophical Society*, 87(1):151–155.

Splechtna, R., Matković, K., Gračanin, D., Jelović, M., and Hauser, H. (2015). Interactive visual steering of hierarchical simulation ensembles. In *2015 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 89–96.

Sun, Y. and Genton, M. G. (2011). Functional boxplots. *Journal of Computational and Graphical Statistics*, 20(2):316–334.

Waser, J., Fuchs, R., Ribicic, H., Schindler, B., Bloschl, G., and Groller, E. (2010). World lines. *IEEE Trans. Visual Comput. Graphics*, 16(6):1458–1467.

Wickelmaier, F. (2003). An introduction to mds. *Sound Quality Research Unit, Aalborg University, Denmark*, 46(5).

Xie, J., Sauer, F., and Ma, K.-L. (2015). Fast uncertainty-driven large-scale volume feature extraction on desktop pcs. In *Large Data Analysis and Visualization (LDAV), 2015 IEEE 5th Symposium on*, pages 17–24. IEEE.