Uncertainty-Aware Exploration of Continuous Parameter Spaces Using Multivariate Prediction

W. Berger[1], H. Piringer[1], P. Filzmoser[2], and E. Gröller[3]

[1]VRVis Research Center, Austria
[2]Department of Statistics and Probability Theory, Vienna University of Technology, Austria
[3]Institute of Computer Graphics and Algorithms, Vienna University of Technology, Austria

Abstract

Systems projecting a continuous n-dimensional parameter space to a continuous m-dimensional target space play an important role in science and engineering. If evaluating the system is expensive, however, an analysis is often limited to a small number of sample points. The main contribution of this paper is an interactive approach to enable a continuous analysis of a sampled parameter space with respect to multiple target values. We employ methods from statistical learning to predict results in real-time at any user-defined point and its neighborhood. In particular, we describe techniques to guide the user to potentially interesting parameter regions, and we visualize the inherent uncertainty of predictions in 2D scatterplots and parallel coordinates. An evaluation describes a real-world scenario in the application context of car engine design and reports feedback of domain experts. The results indicate that our approach is suitable to accelerate a local sensitivity analysis of multiple target dimensions, and to determine a sufficient local sampling density for interesting parameter regions.

Categories and Subject Descriptors (according to ACM CCS):   I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Many key challenges in science and engineering can be generalized as studies of systems denoted as $X \rightarrow Y$, where the parameter space X is a continuous subset of $\mathbb{R}^n$ and the target space Y is a continuous subset of $\mathbb{R}^m$. Examples include model calculations in finance or environmental studies as well as processes in manufacturing and industrial design [TSDS96]. A general goal is to understand the relationship between X and Y in both directions. Conclusions from X to Y often refer to the sensitivity - which changes of which parameters influence which targets in which way? Conclusions from Y to X analyze which regions of the parameter space generate particular results. For many systems, however, each evaluation at a specific point of X incurs the cost of conducting a measurement or running a time-consuming simulation. This explains why in practice an analysis is often restricted to a limited number of samples.

The application background motivating the work of this paper is the development process of car designs by means of 1D-CFD multi-run simulations as described by Matkovic et al. [MJJ*05]. In this context, the parameter space comprises design choices of the engineers (e.g., the timing of fuel injection) as well as conditions varying during the operation (e.g., engine speed). The target space consists of simulated behavior like torque, emission rates, and fuel consumption. In this application context, specific goals of an analysis include the identification of statistical surrogate models in order to approximate time-consuming physical simulation runs in real-time for certain sub-systems of a car like the engine [PBK10]. Such simplifications are necessary to enable studies of the entire system (i.e., the car). Another important task concerns the optimization of design choices in the context of many inherently conflicting objectives [BP10]. The figures in this paper refer to an optimization of a real-world turbo-charged car engine (see Sec. 6.1).

Abstracting from this particular application domain and these tasks, however, many challenges resemble those encountered when studying non-trivial systems in other aforementioned areas. One specific challenge is the data quality. Non-converged simulation runs, for example, generate implausible data samples that must be dealt with before any further steps. Another challenge refers to the density of sample

points. Tasks like model identification and optimization require a sufficient density especially within particular regions of the parameter space. Neither the sufficient density, nor the interesting regions may be known a priori. These challenges motivate an interactive approach that involves the knowledge of domain experts, e.g., to assess the data quality and to locally increase the sampling density if necessary [MGJH08].

This paper describes an interactive approach to support a local analysis of a sampled parameter space. The key idea is to enable a continuous navigation through the parameter space instead of being restricted to a limited number of sample points, and to provide expected results of multiple targets. To generally accomplish this in real-time, the approach employs methods from statistical learning for predicting results at any user-defined point and its neighborhood. In order to ensure flexibility with respect to diverse goals and tasks, we describe both navigation and prediction as a general concept for coordinating multiple views and we provide respective extensions to 2D scatterplots and parallel coordinates.

In particular, the main contributions of this paper address issues of interactive, prediction-based local analysis:

- Providing visual guidance for an efficient navigation to interesting parameter regions in the context of multiple target dimensions and other application-specific criteria.
- Visualizing uncertainty with regards to a potentially insufficient sampling density and inaccurate predictions.
- Evaluating the usefulness based on a real-world application scenario and the user-feedback of domain experts.

2. Related Work

The related work comprises approaches for analyzing parameter spaces and techniques for representing uncertainty.

2.1. Analyzing Parameter Spaces

The analysis of multidimensional parameter spaces has long been a topic of visualization research. Shaffer et al. describe a high dimensional problem for aircraft design [SKW98] which also characterizes the challenges of our application domain very well. As one conclusion, Shaffer et al. postulate "*give designers a feel for objective function and constraint sensitivity relative to the parameter space*".

Many visual approaches for studying complex systems rely on sampling the parameter space of the system. Guo et al., for example, sample in parameter space to identify good linear models that fit given data [GWR09]. In the context of engineering data, Tweedie et al. describe the Influence Explorer and the Prosection Matrix [TSDS96], which employ cross-filtering on multiple 1D and 2D projections. For a similar application, Matkovic et al. [MJJ*05] use linking and brushing in multivariate visualizations. Despite their usefulness, these approaches are limited to pre-computed data samples and provide no information in between.

One solution is to increase the number of sample points

interactively during the analysis. Wright et al. [WBD00] calculate samples on demand and monitor the progress towards a preferred solution while enabling users to navigate a maximum of six design dimensions at once. More recently, Matkovic et al. proposed to generate new sample points by brushing common scatterplots to support visual steering of the simulation [MGJH08]. Their positive user feedback was a motivation for our work. However, in their approach, the user has to wait for the completion of time-consuming simulation runs. We intended to offer the user an approximation of the simulation results in real-time.

Continuous versions of scatterplots [BW08] and parallel coordinates [HW09] have been proposed to generate continuous plots of sampled data. Based on a density function, these techniques map an n-D domain to m-D visualizations. While useful for a global overview, other methods for predicting intermediate results are likely to be superior for a given application. Moreover, it is not possible to infer local details like precise values or the uncertainty of interpolated data. Similar in spirit, Chan et al. proposed flow-based scatterplots [CCM10], which use a local analysis of derivatives to find local trends for sensitivity analysis. However, flow-based scatterplots are restricted to linear regression for approximating the partial derivatives with respect to a single variable. Our approach supports different types of prediction and visualizes the sensitivity of multiple variables.

Some approaches directly visualize high-dimensional scalar functions by preserving global properties after a projection to low-dimensional space. Gerber et al. apply a segmentation of the parameter space of a sampled scalar function [GBPW10]. Subsequent regression generates a low-dimensional representation that preserves local minima and maxima. Based on contour trees [Ree46], Weber et al. propose "Topological Landscapes" as a terrain metaphor for understanding the topological structure [WBP07]. Harvey et al. [HW10] extended this concept to preserve the volume of each topological component in the visualization. While useful to provide an overview of a single function, these approaches are not suitable for a detailed sensitivity analysis with respect to multiple target values.

Many approaches for visualizing scalar functions reduce the number of parameters by slicing. Examples include nested coordinate systems [FB90] and radial layouts of slices [JN02]. HyperSlice shows all 2D orthogonal slices of a function around an n-dimensional focal point in a matrix layout [WL93]. Based on the same concept, HyperMoVal validates regression models in the context of known validation data [PBK10]. This paper continues this research direction in so far as regression is used as one type of prediction method. However, the focus of this paper is not on validating a particular model, but on using multiple models for a continuous exploration of a sampled parameter space.

2.2. Uncertainty Visualization

An effective representation of uncertainty has been recognized as important problem in visualization research [Joh04,

SH10]. As a requirement, some researchers examined what uncertainty is and where it originates from. In their early survey of this topic, Pang et al. [PWL97] distinguish between acquisition, transformation, and visualization as sources of uncertainty. Similarly, Gershon provides a domain-independent taxonomy of imperfect information [Ger98].

Most research on uncertainty visualization has been done in the field of geographic and scientific visualization. Techniques include animation [Ger92], glyphs [LPSW96], annotations [CR00], and modifying geometry [GR04]. MacEachren et al. provide a comprehensive overview of approaches for visualizing geospatial information uncertainty [MRH*05].

For more general data, box plots have long been used in statistics to summarize a distribution of values. There are many extensions to box plots as surveyed by Potter et al. [PKRJ10], who also propose a new hybrid summary plot to represent uncertainty. Olston et al. emphasize the distinction between statistical and inherently bounded uncertainty [OM02]. They argue that a technique called ambiguation should be used to convey bounded uncertainty.

In our context, uncertainty originates from prediction. In their classification, Skeels et al. describe prediction as one of five major sources of uncertainty [SLSR10], but they do not provide information on how to consider this aspect for visualization. As we will discuss in Sec. 4, a visualization of prediction-based uncertainty should take the employed prediction technique into account. This helps users to identify suitable means for reducing the uncertainty. To the best of our knowledge, no research has compared the visualization of uncertainty for different prediction techniques so far.

3. Parameter Space Exploration

This section describes our approach to a guided navigation of multidimensional parameter spaces. First, we generalize the idea of focal points to enable a pointwise navigation and prediction. Based on this concept, Sec. 3.2 then describes techniques for parameter- and target-oriented navigation guides that rely on mapping one space to the other. We describe the techniques for 2D scatterplots and parallel coordinates representing the sample points as these are well-known and widely-used types of visualizations.

3.1. Pointwise Navigation and Prediction

Approaches that rely on slicing high-dimensional spaces have established the notion of a focal point, e.g., Hyper-Slice [WL93] and HyperMoVal [PBK10]. A focal point, denoted as F, is a user-defined n-tuple specifying concrete values $F_i \in \mathbb{R}, i \in \{1 \ldots n\}$ for all n dimensions of the parameter space X. Previous approaches use F to specify axis-orthogonal projections to low-dimensional space for visualization in well-defined small multiple layouts. Our approach recognizes F as a system-wide concept for coordinating multiple views which may be of different type. The
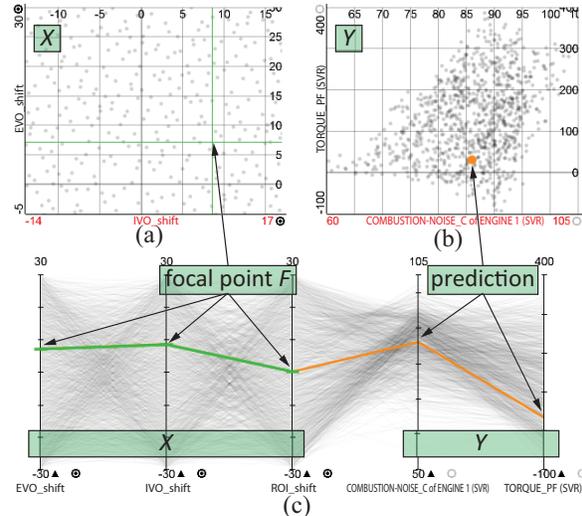

Figure 1: *(a and b) 2D scatterplots represent F as a movable crosshair and the associated prediction as a point; (c) parallel coordinates represent both as connected lines.*

reason for this generalization is increased flexibility regarding possible interactions and visual representations of F.

Different interaction techniques support the specification of F within X. We discriminate global and local updates of F. Local updates affect a subset of dimensions at a time, while global updates define values for all dimensions of X simultaneously. To support local updates, 2D scatterplots projecting dimensions of X represent F as a cross-hair (see Fig. 1a). Moving it modifies F in the two displayed dimensions. Similarly, parallel coordinates draw a movable line at the position of F_i for each axis representing a dimension of X and connect the lines between adjacent axes (see Fig. 1c). Global updates rely on clicking on a visual representation of a sample point, i.e., a point in the 2D scatterplot and a line in parallel coordinates. In case of overlapping items, we disambiguate the selection by taking the sample point which is most similar to the existing values of F in all dimensions.

In order to predict target values, the user specifies a certain prediction technique $\hat{f}_j(X) : \mathbb{R}^n \to \mathbb{R}, j \in \{1 \ldots m\}$ as a surrogate for the more complex real function $f_j(X)$ for each of the m dimensions of the target space Y. To support multiple application contexts, the user can choose between different prediction techniques like regression models or k-nearest neighbor estimators (see also Sec. 4). Each modification of F instantaneously triggers an evaluation of all predictions for the new values of F. The result is conceptually similar to any given sample point and can thus be visualized likewise in plots showing the target space, i.e., as a point in scatterplots and a line-strip in parallel coordinates (see Fig. 1b, c). Considering the different semantics, though, we use color to visually discriminate predictions from samples. Even without extensions, the immediate feedback on Y while navigating X can significantly support a sensitivity analysis.

3.2. Guided Navigation

While the interaction concept as described in Sec. 3.1 is very flexible, it has two shortcomings. First, the analysis is restricted to a single point at a time, and second, the user may easily get lost in the parameter space. To address both issues, this section describes extensions to consider a local neighborhood around F which also provides a guidance towards specific characteristics of Y. A neighborhood around F may either be defined with respect to X or Y. Our approach supports both options and maps the neighborhood to the respective other space in order to achieve different goals:

- Mapping $X \rightarrow Y$ indicates predicted target ranges that are within reach by varying F (Sec. 3.2.1).
- Mapping $Y \rightarrow X$ shows the predicted sensitivities of multiple targets to changes of F (Sec. 3.2.2).

3.2.1. Mapping $X \rightarrow Y$

The definition of a local neighborhood with respect to X is based on a set of normalized n-dimensional variation vectors. This set of vectors is denoted as V, with $|V_i| \leq r$, a user-defined radius to control the degree of locality. After scaling all elements of V to match the extents of X, adding F to V defines a set of variation points $P_V(F)$ around F. The main idea is to apply prediction models to all elements of $P_V(F)$ and to represent them like additional data items in visualizations of X as well as of Y. We use color to distinguish $P_V(F)$ from F and from the original data and to provide information about a particular variation vector V_i. As a general guideline, hue encodes dimensions of X while luminance is modulated to represent the magnitude of V_i. However, details depend on the strategy for creating V as described later.

The visualization of $P_V(F)$ indicates target ranges within reach of F and it also enables a quick navigation to interesting points. In consistence with the concept of global updates of F, the user may click on any variation point to set all dimensions of F accordingly. For overlapping data items, points of $P_V(F)$ are preferred over items of the original data, and similarity to F (i.e., the magnitude of the vectors) is used for further disambiguation. Repeated updates of F enable to explore a large number of neighborhoods within a short time. This interaction is thus an efficient way to locally analyze a system while steering towards certain characteristics.

The information provided for the neighborhood and the amount of clutter largely depend on the strategy for defining V. One may conceive multiple strategies for defining V. We currently provide two strategies aiming at different goals.

One strategy to define V is to vary a single dimension at a time in steps of size s, which yields a set of $n*(2r/s)$ vectors in total. Due to the visual appearance, we refer to that strategy as **star sampling** (see Fig. 2). Concerning the coloring scheme, hue is defined by the varied dimension. Positive variations slightly blend this hue against white with increasing magnitude, and negative variations against black. In 2D scatterplots, lines connecting points provide an additional vi-
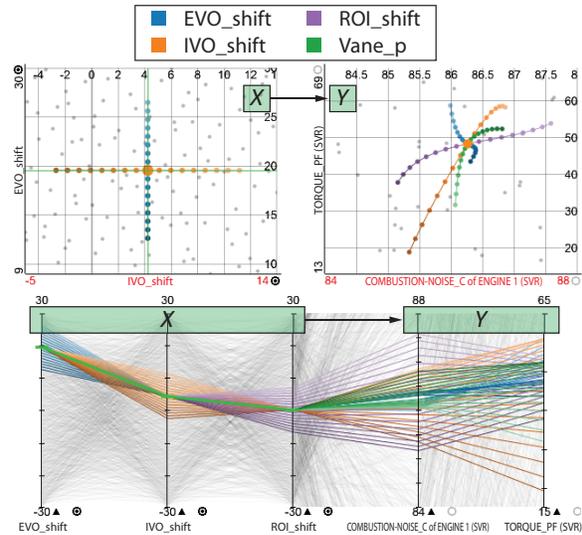

Figure 2: *Star sampling varies a single parameter at a time. The mapping to Y shows a high local sensitivity of torque to IVO_shift. Combustion noise mostly depends on ROI_shift.*

sual cue about relations between variations. More specifically, all points are connected which are varied in the same dimension in the order of their variation. A comparison of the resulting trails in visualizations of X and Y provides details of the combined sensitivity of two target values with respect to individual variations of all parameters. Despite this advantage and its simplicity, star sampling suffers from two drawbacks. First, it generates an increasing amount of clutter for an increasing dimensionality of X, which limits its scalability to higher-dimensional parameter spaces. Second, it does not capture interactions between multiple parameters which are a key aspect in practice.

To overcome these restrictions, we provide **stochastic sampling** as an alternative strategy to define V (see Fig. 3). We generate a user-defined number of sample points which are equally distributed inside an n-dimensional hypersphere around F with radius r [CDW96]. Stochastic sampling captures effects of varying multiple parameters at a time, and scales to a high dimensionality of X. However, these advantages come at the cost of losing an intuitive relationship between particular variations in X and their projections to Y. Color helps to mitigate this problem. As the limited color space cannot describe a variation in all n dimensions, we assign each point the hue of the dimension in which V_i has the absolute largest variation. In consistence with star sampling, the sign and the magnitude of $P_{V_i}(F)$ are indicated by a slight blending against black or white. This coloring scheme enables the identification of parameters with a significant impact on Y, but cannot fully describe V. Therefore, neither star sampling nor stochastic sampling are better in general, but they are two alternatives with complementary advantages and disadvantages.
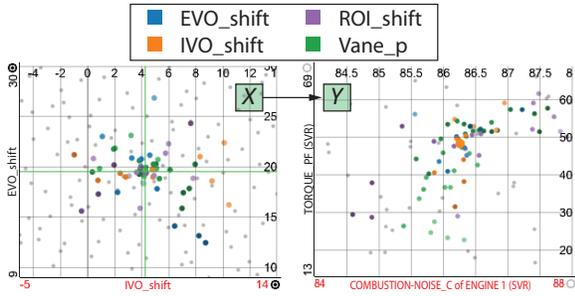
Figure 3: *Stochastic sampling varies multiple parameters at once. The distribution in Y suggests a small local impact of EVO_shift compared to ROI_shift and Vane_p.*

3.2.2. Mapping $Y \rightarrow X$

A direct mapping from Y to X is generally not possible. This implies that an evaluable neighborhood of F with respect to Y still has to be defined in terms of X. Our key idea is to visualize neighborhoods around F where predictions differ from those at F by at most a certain constant value. We define a separate neighborhood for each dimension of Y. The neighborhood of F with respect to the k-th dimension of Y is defined as a contiguous region $R_k \subseteq X$ around F where $|f_k(\Phi) - f_k(F)| < \varepsilon \ \forall \ \Phi \in R_k$. For visualization, the number of variable dimensions of R_k may be smaller than n while assuming the values of F for all other dimensions.

In our case, we visualize R_k within a 2D scatterplot projection of X_i, X_j and thus choose R_k to be two-dimensional. Conceptually, the visualization of R_k corresponds to the area between the isolines at $\hat{f}_k(F) - \varepsilon$ and $\hat{f}_k(F) + \varepsilon$ of the 2D function obtained by slicing $\hat{f}_k(F)$ at F for all dimensions of X except X_i and X_j. This area indicates how F_i, F_j can vary without changing the prediction by more than ε. Its shape provides important information (see Fig. 4): The width in any direction is inversely proportional to the sensitivity of $\hat{f}_k(F)$ with respect to combined variations of X_i, X_j in that direction. Furthermore, an area forming a closed shape indicates the presence of a local extremum close to F.

We encode a specific target dimension by the hue of its area. Altering the luminance discriminates sub-areas where predictions lie above or below $\hat{f}_k(F)$. The boundary between these sub-areas denotes the isoline of $\hat{f}_k(F)$ at F. This is useful information and explains why we do not apply a smooth gradient in luminance. Our experience indicates that filling the area clearly conveys what is inside. Drawing only outlines can be misleading in this respect.

Our goal is to convey the sensitivity with respect to multiple dimensions of Y simultaneously. A 2D scatterplot may thus contain multiple areas. To reduce the amount of occlusion, the areas are ordered by the magnitude of their maximal gradient at F in any 2D direction within X_i, X_j. This ensures that highly sensitive target dimensions are always visible and it also draws thin areas on top of broader areas so that little information is lost in general. To ensure that no information
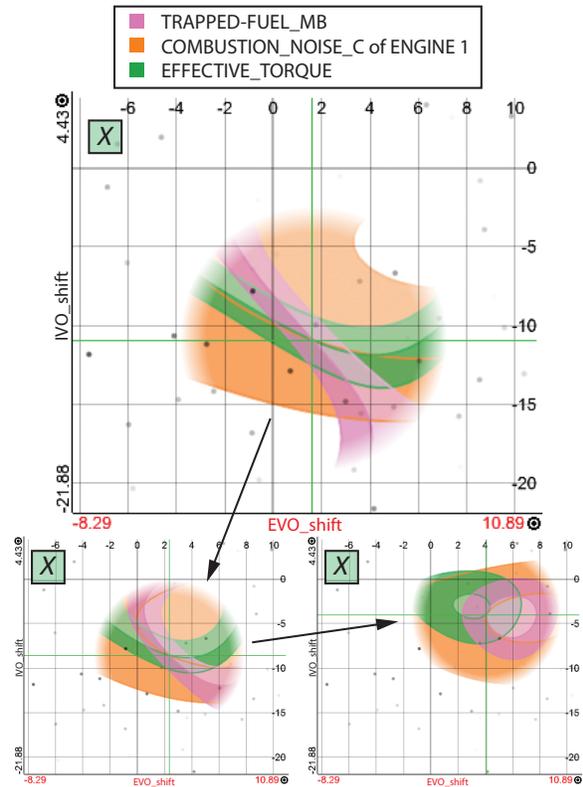

Figure 4: *Mapping the ε neighborhood of three targets into X shows their sensitivities to combined changes of IVO_shift and EVO_shift. Increasing both parameters locally minimizes trapped fuel but also torque.*

is lost at all, additional lines outline the borders of each area. Alternatively, the user may also manually specify an order to focus on particular targets.

Drawing multiple areas in 2D scatterplots visualizes the sensitivities and trade-offs between different targets with respect to combined variations of two parameters. This has proven powerful for multiple purposes (see Sec. 6), but has two limitations concerning scalability. First, experience shows that a practical limit of visualized target dimensions is around five. Larger numbers are both visually and mentally too complex. We thus allow users to choose a subset of target dimensions for visualization. This also enables users to open a second plot for the same parameters showing different targets.

The second limitation was identified by user feedback and refers to the amount of visual change while navigating F. As a region R_k depends on the value of the prediction $\hat{f}_k(F)$, it changes whenever the user moves F. Drawing the entire projection of R_k generates much visual change also far away from F as the center of attention. This has been considered very distracting when drawing multiple areas. As a consequence, we decided to draw areas only within a fixed screen space radius around F. This "spyhole" metaphor reduces the

visual complexity and noise while providing the same information for the neighborhood. It also emphasizes the purpose as a local guide. A tooltip displays the absolute predictions at the current mouse cursor and their differences relative to the predictions of F. This supports an instantaneous examination of predictions both inside and outside the spyhole, and provides textual detail on demand.

The reason for embedding the areas within a scatterplot is to provide the distribution of known sample points as context. This allows for setting F to any of these points by a global update, and it also reveals the local sampling density. However, the 2D projection to (X_i, X_j) does not convey the real distance of a sample from F within X in all dimensions. Therefore, we optionally support a similar concept as described for HyperMoVal [PBK10], as we increase the transparency of a sample point with increasing distance in the remaining dimensions of X except X_i and X_j.

4. Uncertainty of Predictions

All projections from X to Y of Sec. 3.2 are based on prediction methods $\hat{f}_j(F)$ as surrogates for the more complex real function $f_j(F)$. Even good predictions will in general deviate from the true functions. For a full picture, it is necessary to convey this prediction uncertainty.

In our system, users may choose between prediction methods for each target dimension. Different prediction methods, however, have different sources of uncertainty. Regression models typically make significant assumptions about the structure and may thus fail to sufficiently reflect the complexity of the respective target. Nearest-neighbor (NN) predictors make only mild structural assumptions, but depend directly on the local sampling density [HTF09]. We therefore distinguish between NN and model-based predictors and describe how uncertainty can be determined, visualized and handled for these types.

4.1. Uncertainty of Nearest-Neighbor Predictors

NN predictors assign a weight $w_l, l \in \{1 \ldots k\}, \sum_{l=1}^{k} w_l = 1$ to those k samples, which have the smallest normalized distance to F in X. The predicted target at F is a weighted sum of the target values. Weights depend on F and must be recomputed at each update.

Assuming a well-chosen value of k based on a preceding training, the prediction uncertainty depends on the distance to the selected samples and their variance with regard to the associated target values. A high variance indicates a significant local gradient. The amount of uncertainty introduced by that gradient depends on the distribution of the samples in X with regard to F. We thus quantify the uncertainty of an NN predictor in terms of the weighted distribution characteristics of the k nearest neighbors with regard to Y. As characteristics, we compute the well-known box plot measures minimum, maximum and weighted variants of the lower and upper quartile as well as the median.
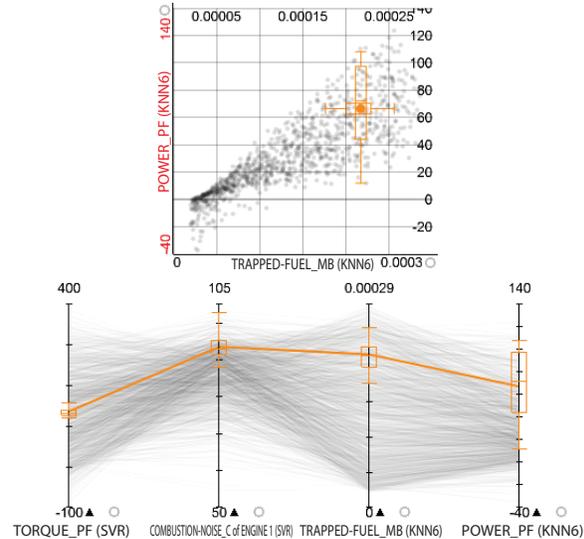

Figure 5: *Box plots uniformly visualize uncertainties of model- and NN-based predictors. For POWER_PF, a wide spread indicates a highly uncertain prediction.*

4.2. Uncertainty of Model-Based Predictors

A model-based supervised prediction method uses a set of training data to infer an internal representation, e.g., a hyperplane in case of linear regression. This representation enables the model to predict a certain target dimension for any point of X. Building such a model is a task on its own and beyond the scope of this work [HTF09, PBK10].

In contrast to NN predictors, a model-based predictor is independent of concrete data samples after its training and validation. However, the residuals at known data points are suitable to assess the local prediction accuracy. A local distribution of residuals characterizes both the standard deviation (i.e., the average prediction error), as well as a local tendency to over- or underestimate results. Both types of information represent a specific kind of uncertainty. To compute the distribution of residuals around F, we employ a weighted NN prediction at F on these residuals. This obtains the summary statistics of a box plot (see Sec. 4.1). Being computed on residuals, these measures are relative rather than absolute values. By adding $\hat{f}_j(F)$, we are able to express the local prediction accuracy as a distribution in Y.

4.3. Visual Encoding

Although the two types of predictors have different sources of uncertainty, it can be described in terms of a distribution within Y in both cases. We use well-known box plots to convey the uncertainty for both types of predictions in visualizations representing Y (see Fig. 5). The 2D scatterplot presents them as overlays at the predicted point while the parallel coordinates visualize them as axis overlays. Wide box plots generally represent a high uncertainty at F. For model-based predictors, a shift to either side of the predicted result indicates a local tendency to over- or under-estimation. We thus
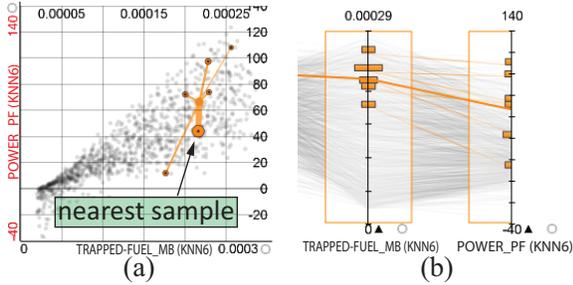
Figure 6: *Visualizing NN-based predictors in terms of sample weights. The used samples show a high variance for POWER_PF. The 2D scatterplot also indicates a significant deviation from the nearest sample in X having most weight.*

represent uncertainty in a uniform way, while the labels of axes inform the user about the employed prediction method.

NN predictors directly rely on the sample data. An additional idea is to relate their predictions to the underlying sample points. The purpose of this visualization is to make users aware that neighboring points in X may be distributed all over Y (see Fig. 6) and to represent the forces that attract the predicted value towards particular sample points.

In 2D scatterplots showing Y, lines connect the predicted point to the k samples (Fig. 6a). For each line, its width and the point size of the connected sample represent the relative weight with respect to the remaining $k - 1$ samples. This enables a quick identification of involved sample points and their individual influence on the prediction.

In parallel coordinates, a histogram visualizes the distribution of the k samples on the axis of each target dimension that is predicted by a NN estimator (see Fig. 6b). The height of each bin corresponds to the sum of the weights of samples within the bin. We decided for a binned representation to avoid problems due to overlap in case of a narrow distribution which is a frequent case for good predictions.

In case of a high uncertainty, a predicted value may be regarded as useless (e.g., the prediction of POWER_PF in Fig. 6). Adding samples around F can locally reduce the uncertainty of NN-based predictions. Reducing the uncertainty of model-based predictors is not as straightforward. Unless it is possible to accept the uncertainty as inherent, the user may either attempt to improve the model (e.g., re-train it for different parameters), or switch to an NN-predictor and ensure a sufficient local sampling density. This second option was also a motivation to pay special attention to NN.

5. Implementation

The described approach has been implemented within *visplore*, a system for visual exploration based on multiple linked visualizations. It is written in C++, uses OpenGL for rendering and GTK+ for GUI elements.

A system-core is responsible for view coordination. The core centrally stores prediction results and updates them on changes of F. Clients (e.g., views) register to be no-

tified about updates of F and of prediction results. Multi-threading ensures the responsiveness of the application during updates and provides early visual feedback when moving F [PTMB09]. All prediction methods implement a common interface for updating and accessing results and uncertainty information. Our implementation currently provides support vector regression based on the library LIBSVM [CL] as well as a weighted NN-based predictor. An open architecture makes adding new prediction methods or different strategies for defining variation vectors straightforward.

6. Evaluation

We evaluate our approach on two levels. First, we present an application scenario which describes the integration of our approach into a typical workflow in the field of car-engine design. Furthermore, we conducted interviews to collect feedback from five experts in car engine-design, who have been testing our approach within their domain. The gathered insights form the second part of our evaluation.

6.1. Application Example

The goal of this example is to optimize the design of a real-world turbo-charged car engine. There are two conflicting major objectives: Maximize the engine's torque and minimize its fuel consumption. A third minor objective is to keep the combustion noise as low as possible. This example focuses on the engine's cruising mode, i.e., a load state of about 20 % and average engine speed of 2000 rotations per minute. The analysis is based on results from 1000 simulation runs which stochastically sampled a six-dimensional parameter space. Load signal and engine speed are operating parameters (OPs) as they vary during the operation of the engine. The rest are design parameters (DPs) which are specified by the engineer: Vane_p (turbine vane position of the turbocharger), IVO_shift (intake-valve opening-shift), EVO_shift (exhaust-valve opening-shift), and ROI_shift (relative shift of the start of injection). Support vector regression-based predictors for the targets have been identified and validated in a preceding step.

The first goal is to identify an appropriate starting point for the analysis. An integrated pareto optimization (see [BP10]) selects candidate samples with respect to the three objectives. Further refinement restricts the selection to samples with load signals within $(10, 30)$ % and engine speeds within $(1800, 2200)$ rpm (see Fig. 7b). The engineer sets F to the candidate sample that is closest to the center of the analyzed region (Fig. 7a, b).

Star sampling highlights the different impact of the DPs regarding the two major objectives. In a 2D scatterplot for torque and trapped fuel (Fig. 7c), the purple trail representing ROI_shift contains variation points which improve on both. The engineer sets F to the best variation point. Updated variations show no potential for further improving both objectives, but the trail of ROI_shift indicates the possibility to trade a small amount of torque for a comparatively high gain
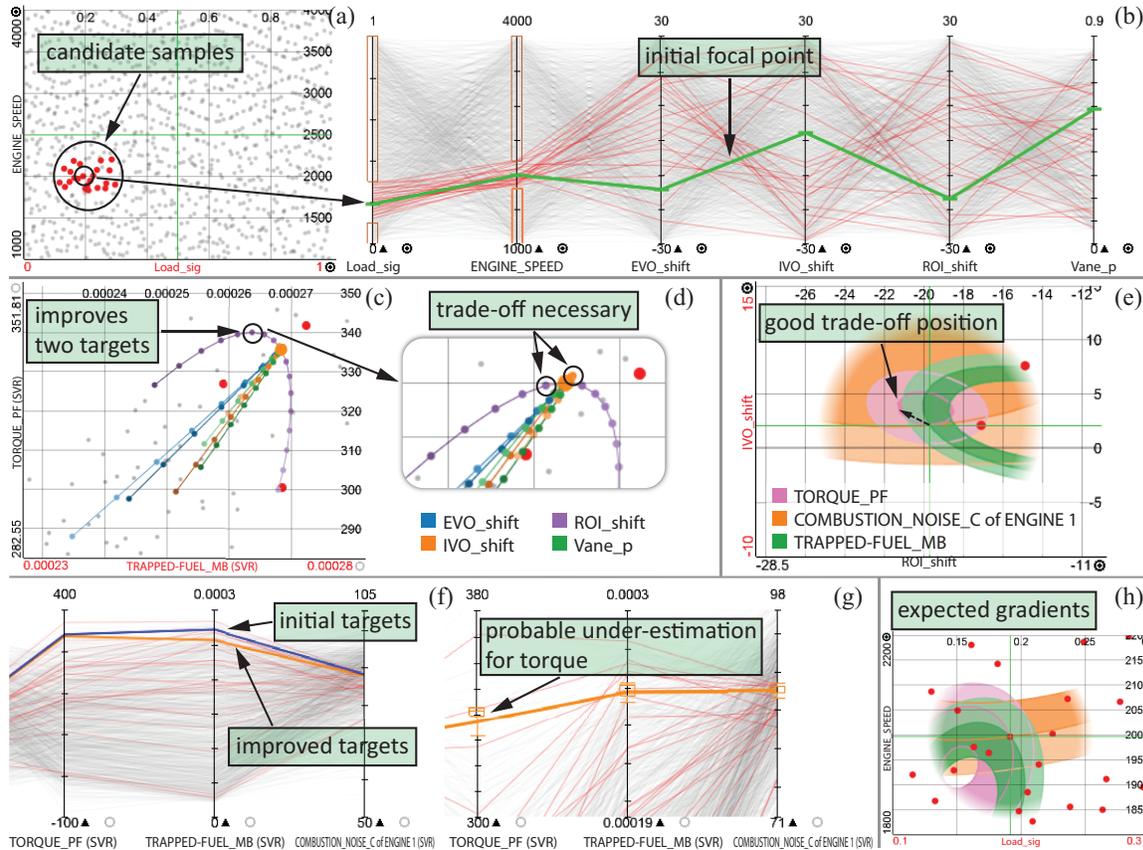
Figure 7: *Exemplary workflow. (a) Clicking a candidate sample sets F to the values shown in (b). (c) Parameter variation reveals an improvement for torque and fuel consumption. (d) Variations at the new F indicate a possibility to trade torque against lower trapped fuel. (e) Changing IVO_shift and ROI_shift reduces trapped fuel, keeps torque constant and slightly increases noise. (f) Comparing the target predictions to the initial values. (g) The uncertainty box plot for torque indicates a probable under-estimation. (h) Areas show a reasonable behavior of all targets for changing operating parameters.*

regarding fuel conservation (Fig. 7d). Moreover, the variations of IVO_shift highlight an option to further increase torque at the cost of a higher fuel consumption.

To investigate these trade-offs in more detail, a 2D scatterplot for ROI_shift and IVO_shift shows area guides for torque, trapped fuel and noise (Fig. 7e). The visualization reveals that increasing IVO_shift and decreasing ROI_shift can reduce the fuel consumption without changing torque at the cost of slightly higher combustion noise. The tooltip confirms a possible 1.4 % decrease of fuel consumption, no change in torque and a 0.3 % increase of noise. At the new position, no reasonable further trade-offs suggest to stop the parameter search at this point. Compared to the start sample, a decrease of 4.3 % in fuel consumption has been achieved, while torque and noise worsened only by 0.3 %, as shown in the parallel coordinates of Fig. 7f.

Before simulating additional samples around *F*, the engineer wants to ensure that the predicted values are trustworthy. The box plots for noise and fuel consumption indicate good predictions with a low uncertainty (Fig. 7g). The box

plot for torque reveals a slight tendency of the predictor to underestimate values around *F*, which suggests that a simulation run at *F* might have even better torque characteristics.

In a final step the engineer investigates the sensitivity of the identified design with regard to changing OPs. In a 2D scatterplot for load signal and engine speed, the area guides for torque, noise and trapped fuel indicate gradients which are plausible and expected for the investigated engine (Fig. 7h). This convinces the engineer to add simulated samples in the region around *F* in order to confirm the findings.

6.2. User feedback

According to the interviewed domain experts, the current workflow for analyzing and optimizing engines is a combination of automated methods and trial & error. The design process involves not only finding a single set of optimal parameters, but also investigating its surroundings, as slight fluctuations during production should not cause disproportional performance changes. Early feedback from engineers reports that the immediate prediction for pointwise

navigation helps to convey the dynamics of the investigated systems regarding parameter changes. In general, they consider the guiding mechanisms as very helpful to quickly navigate to interesting points and to get an impression of the sensitivity of the simulated system. One engineer noted that he prefers star sampling to stochastic sampling due to its regular appearance which makes projections easier to interpret. Other feedback motivated the spyhole metaphor for the area-based guides (Sec. 3.2.2). The engineers regard the uncertainty of predictions as crucial information that helps to justify expensive additional simulation runs. NN-based prediction is applied when model-based prediction is not sufficiently accurate. In this case, interviewees considered the weight visualization as helpful to assess the plausibility of predictions in more detail than conveyed by box plots.

An interviewed domain expert emphasized that our approach has high relevance within the context of car engineering. Recent developments like hybrid-engines have multiple complex car components interacting with each other which introduces a high degree of freedom when defining design parameters. Current workflows need adaptation to handle this increased complexity. The expert thus expects the presented approach to be an important part of new workflows.

7. Discussion and Future Work

The described approach has been designed with three main goals in mind: 1) Investigate system dynamics by analyzing the local sensitivity of multiple target dimensions at a time, 2) examine the local precision and plausibility of predictors, and 3) support assessments for which regions the sampling density should be increased. The latter issue is closely related to a statistical problem known as design of experiments (DOE). For many real-world systems, a purely statistical approach to DOE would be prohibitively complex or inaccurate. We thus consider our approach as an important contribution to an interactive solution. Being local by definition, the choice of an appropriate starting point for F is crucial for many tasks. Interaction techniques like brushing samples and computed information like pareto frontiers [BP10] may complement knowledge of a domain expert to identify candidates for starting points (see Sec. 6.1). As compared to global approaches, a local analysis is inherently restricted to a small region of the parameter space at a time but it enables to display more details for that region.

Visual guides are based on predictions and may thus differ slightly depending on the chosen prediction method. This is one reason why we consider it important to provide different prediction methods and let the user change the employed method at run-time. The information about local uncertainty may support this choice. In our implementation, the user may enable one type of guide or uncertainty visualization per view. For example, a scatterplot of two parameters may show area-based guides for target dimensions, while linked parallel coordinates provide precise prediction results and uncertainty information for the targets.

The visual scalability of our approach is conceptually independent of the number of sample points. As discussed in Sec. 3.2.2, we identified a practical upper limit of about five visualized target dimensions for area-based guides in a single plot. The scalability of mappings from X to Y (Sec. 3.2.1) depends on the sampling strategy rather than on the dimensionality of X or Y. However, the use of color to distinguish dimensions visually suggests a practical upper limit of 12 simultaneously displayed dimensions for the navigation guides [War04]. Concerning computational scalability, all mentioned prediction methods are fast enough to support hundreds of evaluations per second. In combination with multi-threading and the use of graphics hardware, our approach provides visual feedback in real-time.

As future work, we plan to create visual guides for other visualizations, e.g., parallel coordinates and 3D scatterplots, based on the general definition of the local neighborhood R_k as described in Sec. 3.2.2. Furthermore, gradient-based concepts could support a semantically meaningful navigation of F in all dimensions of X simultaneously. We also intend to further evaluate our approach within other application domains. Finally, we intend to extend our concept to categorical parameters as well as results. Based on classification methods for prediction, categorical results require different visualization concepts for guides and uncertainty.

8. Conclusion

This paper introduced an interactive approach to a continuous analysis of a sampled parameter space. The concept of a system-wide focal point enables a local exploration in linked views of different type. We employ statistical methods to predict local results for multiple target dimensions in real-time. Mapping a local neighborhood with respect to the parameter or target space to the respective other space guides the user to interesting target ranges and visualizes the sensitivities of multiple targets to parameter changes. Embedding these guides within 2D scatterplots or parallel coordinates provides known sample points as important context information. We discussed different sources of inherent uncertainties of model-based and nearest-neighbor-based methods and described approaches to visualize them. User feedback indicates that our approach increases the confidence in gained insights as it helps to understand the dynamics of the investigated systems and supports an assessment of an appropriate sampling density. Experts in the field of car engineering see great importance of this approach with regard to the rising complexity of new challenges within their domain. We believe that our approach may be of general importance in science and engineering.

9. Acknowledgments

References

[BP10]   BERGER W., PIRINGER H.: Interactive Visual Analysis of Multiobjective Optimizations. In *Proc. of the IEEE Conference on VAST 2010* (2010), pp. 215–216. 1, 7, 9

[BW08]   BACHTHALER S., WEISKOPF D.: Continuous Scatterplots. *IEEE Trans. on Visualization and Computer Graphics 14*, 6 (2008), 1428–1435. 2

[CCM10]   CHAN Y.-H., CORREA C. D., MA K.-L.: Flow-based Scatterplots for Sensitivity Analysis. In *Proc. of the IEEE Conference on VAST 2010* (2010), pp. 43–50. 2

[CDW96]   CUMBUS C., DAMIEN P., WALKER S.: Uniform Sampling in the Hypersphere via Latent Variables and the Gibbs Sampler. 1996. 4

[CL]   CHANG C.-C., LIN C.-J.:   LIB-SVM. http://www.csie.ntu.edu.tw/~cjlin/libsvm/, Last visited on Mar. 3rd 2011. 7

[CR00]   CEDILNIK A., RHEINGANS P.: Procedural Annotation of Uncertain Information. In *Proc. of the IEEE Conference on Visualization 2000* (2000), pp. 77–83. 3

[FB90]   FEINER S. K., BESHERS C.: Worlds Within Worlds: Metaphors for Exploring N-Dimensional Virtual Worlds. In *Proc. of the 3rd annual ACM SIGGRAPH Symposium on User Interface Software and Technology* (1990), pp. 76–83. 2

[GBPW10]   GERBER S., BREMER P.-T., PASCUCCI V., WHITAKER R.: Visual Exploration of High Dimensional Scalar Functions. *IEEE Trans. on Visualization and Computer Graphics 16*, 6 (2010), 1271–1280. 2

[Ger92]   GERSHON N. D.: Visualization of Fuzzy Data Using Generalized Animation. In *Proc. of the IEEE Conference on Visualization 1992* (1992), pp. 268–273. 3

[Ger98]   GERSHON N. D.: Visualization of an Imperfect World. *IEEE Computer Graphics and Applications 18*, 4 (1998), 43–45. 3

[GR04]   GRIGORYAN G., RHEINGANS P.: Point-Based Probabilistic Surfaces to Show Surface Uncertainty. *IEEE Trans. on Visualization and Computer Graphics 10*, 5 (2004), 564–573. 3

[GWR09]   GUO Z., WARD M. O., RUNDENSTEINER E. A.: Model Space Visualization for Multivariate Linear Trend Discovery. In *Proc. of the IEEE Symposium on VAST 2009* (2009), pp. 75–82. 2

[HTF09]   HASTIE T., TIBSHIRANI R., FRIEDMAN J.: *The Elements of Statistical Learning*, 2nd ed. Springer, 2009. 6

[HW09]   HEINRICH J., WEISKOPF D.: Continuous Parallel Coordinates. *IEEE Trans. on Visualization and Computer Graphics 15*, 6 (2009), 1531–1538. 2

[HW10]   HARVEY W., WANG Y.: Topological Landscape Ensembles for Visualization of Scalar-Valued Functions. *Computer Graphics Forum 29*, 3 (2010), 993–1002. 2

[JN02]   JAYARAMAN S., NORTH C.: A Radial Focus + Context Visualization for Multi-Dimensional Functions. In *Proc. of the IEEE Conference on Visualization 2002* (2002), pp. 443–450. 2

[Joh04]   JOHNSON C.: Top Scientific Visualization Research Problems. *IEEE Computer Graphics and Applications 24*, 4 (2004), 13–17. 2

[LPSW96]   LODHA S. K., PANG A., SHEEHAN R. E., WITTENBRINK C. M.: UFLOW: Visualizing Uncertainty in Fluid Flow. In *Proc. of the IEEE Conference on Visualization 1996* (1996), pp. 249–254. 3

[MGJH08]   MATKOVIĆ K., GRACANIN D., JELOVIĆ M.,

HAUSER H.: Interactive Visual Steering - Rapid Visual Prototyping of a Common Rail Injection System. *IEEE Trans. on Visualization and Computer Graphics 14*, 6 (2008), 1699–1706. 2

[MJJ*05]   MATKOVIĆ K., JELOVIĆ M., JURIĆ J., KONYHA Z., GRAČANIN D.: Interactive Visual Analysis and Exploration of Injection Systems Simulations. In *Proc. of the IEEE Conference on Visualization 2005* (2005), pp. 391–398. 1, 2

[MRH*05]   MACEACHREN A. M., ROBINSON A., HOPPER S., GARDNER S., MURRAY R., GAHEGAN M., HETZLER E.: Visualizing Geospatial Information Uncertainty: What We Know and What We Need to Know. *Cartography and Geographic Information Science 32*, 3 (July 2005), 139–160. 3

[OM02]   OLSTON C., MACKINLAY J.: Visualizing Data with Bounded Uncertainty. In *Proc. of the IEEE Symposium on Information Visualization 2002* (2002), pp. 37–40. 3

[PBK10]   PIRINGER H., BERGER W., KRASSER J.: Hyper-MoVal: Interactive Visual Validation of Regression Models for Real-Time Simulation. *Computer Graphics Forum 29*, 3 (2010), 983–992. 1, 2, 3, 6

[PKRJ10]   POTTER K., KNISS J., RIESENFELD R., JOHNSON C. R.: Visualizing Summary Statistics and Uncertainty. *Computer Graphics Forum 29*, 3 (2010), 823–832. 3

[PTMB09]   PIRINGER H., TOMINSKI C., MUIGG P., BERGER W.: A Multi-Threading Architecture to Support Interactive Visual Exploration. *IEEE Trans. on Visualization and Computer Graphics 15*, 6 (2009), 1113–1120. 7

[PWL97]   PANG A. T., WITTENBRINK C. M., LODHA S. K.: Approaches to Uncertainty Visualization. *The Visual Computer 13*, 8 (Nov. 1997), 370–390. 3

[Ree46]   REEB G.: Sur les points singuliers d'une forme de Pfaff complètement intégrable ou d'une fonction numérique. *Comptes Rendus Acad. Sciences 222* (1946), 847–849. 2

[SH10]   SANTUCCI G., HAUSER H.: Data Management - Challenges and Opportunities. In *Mastering the Information Age Solving Problems with Visual Analytics*, Keim D., Kohlhammer J., Ellis G., Mansmann F., (Eds.). Eurographics Association, 2010, pp. 32–36. 2

[SKW98]   SHAFFER C. A., KNILL D. L., WATSON L. T.: Visualization for Multiparameter Aircraft Designs. In *Proc. of the IEEE Conference on Visualization 1998* (1998), pp. 491–494. 2

[SLSR10]   SKEELS M., LEE B., SMITH G., ROBERTSON G. G.: Revealing Uncertainty for Information Visualization. *Information Visualization 9*, 1 (2010), 70–81. 3

[TSDS96]   TWEEDIE L., SPENCE R., DAWKES H., SU H.: Externalising Abstract Mathematical Models. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems* (1996), pp. 406–412. 1, 2

[War04]   WARE C.: *Information Visualization: Perception for Design*, 2nd ed. Morgan Kaufmann Publishers Inc., 2004. 9

[WBD00]   WRIGHT H., BRODLIE K., DAVID T.: Navigating High-Dimensional Spaces to Support Design Steering. In *Proc. of the IEEE Conference on Visualization 2000* (2000), pp. 291–296. 2

[WBP07]   WEBER G., BREMER P.-T., PASCUCCI V.: Topological Landscapes: A Terrain Metaphor for Scientific Data. *IEEE Trans. on Visualization and Computer Graphics 13*, 6 (2007), 1416–1423. 2

[WL93]   WIJK J. J. V., LIERE R. V.: HyperSlice: Visualization of Scalar Functions of Many Variables. In *Proc. of the IEEE Conference on Visualization 1993* (1993), pp. 119–125. 2, 3