Honda Research Institute EU
**Honda Research Institute Europe GmbH**
https://www.honda-ri.de/

# Shape Mining: A Holistic Data Mining Approach for Engineering Design Synthesis

## Lars Gräning, Bernhard Sendhoff

### 2014

**Preprint:**

This is an accepted article published in Advanced Engineering Informatics. The final authenticated version is available online at: https://doi.org/[DOI not available]

Advanced Engineering Informatics xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei

# Shape mining: A holistic data mining approach for engineering design

Lars Graening *, Bernhard Sendhoff

*Honda Research Institute Europe GmbH, Offenbach, Germany*

ARTICLE INFO

*Article history:*
Received 17 March 2013
Received in revised form 17 January 2014
Accepted 7 March 2014
Available online xxxx

*Keywords:*
Computer aided engineering
Data mining
Unified design representation
Design concepts
Sensitivity & interaction analysis
Passenger car design

ABSTRACT

Although the integration of engineering data within the framework of product data management systems has been successful in the recent years, the holistic analysis (from a systems engineering perspective) of multi-disciplinary data or data based on different representations and tools is still not realized in practice. At the same time, the application of advanced data mining techniques to complete designs is very promising and bears a high potential for synergy between different teams in the development process. In this paper, we propose shape mining as a framework to combine and analyze data from engineering design across different tools and disciplines. In the first part of the paper, we introduce unstructured surface meshes as meta-design representations that enable us to apply sensitivity analysis, design concept retrieval and learning as well as methods for interaction analysis to heterogeneous engineering design data. We propose a new measure of relevance to evaluate the utility of a design concept. In the second part of the paper, we apply the formal methods to passenger car design. We combine data from different representations, design tools and methods for a holistic analysis of the resulting shapes. We visualize sensitivities and sensitive cluster centers (after feature reduction) on the car shape. Furthermore, we are able to identify conceptual design rules using tree induction and to create interaction graphs that illustrate the interrelation between spatially decoupled surface areas. Shape data mining in this paper is studied for a multi-criteria aerodynamic problem, i.e. drag force and rear lift, however, the extension to quality criteria from different disciplines is straightforward as long as the meta-design representation is still applicable.
© 2014 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-SA license (http://creativecommons.org/licenses/by-nc-sa/3.0/).

## 1. Introduction

The intensive use of computational engineering tools in the recent years and the transition from an experiment to a simulation based product design process, in particular in the automotive industry, has led to a significant increase of computer-readable design data relating design characteristics[1] to the design quality.[2]

In the context of Product Data Management (PDM) and Product Lifecycle Management (PLM), product related data is maintained and integrated through the whole design process or even through the whole lifetime of the product. Although PDM/PLM frameworks have been successful in managing CAD models and documents as well as in integrating CAD and ERP (Enterprise Resource Planning) systems, PLM solutions still need customization to the actual tools used in the design process [1]. Furthermore, the handling of multi-disciplinary processes, tools and data structures as well as a systems engineering or holistic interpretation of the design process remains to be challenging, e.g. see [1,2]. Industrial informatics in the domain of PDM and PLM still has not received the required attention in the literature, e.g. see [3]. As a result the application of data mining techniques to engineering data in practice is still often restricted to single design processes and individual design teams working on a certain CAE task, which we will call a sub-process in the following. The stronger the variation between the CAE tasks is (different representations, different disciplines, different tools and data structures), the more isolated is the data handling. Even though the data might be integrated into an overall PDM framework, it is not available for a holistic data mining approach from a systems engineering perspective. As a simple example different design teams might focus on the aerodynamics of the frontal part of the car, the rear part of the car, the noise generated or the cooling of the front brakes. However, the CAE results as well as the changes the teams proposed to the design are seldom independent from each other, since they are very likely to employ different representations of the design parts. This makes it difficult for data mining techniques to integrate data across teams and disciplines. On the one hand, the decomposition of the overall design problem (known as Simultaneous or Concurrent Engineering) is necessary

* Corresponding author. Tel.: +49 (0)69 8 90 11 750.
*E-mail address:* lars.graening@honda-ri.de (L. Graening).
*URL:* http://www.honda-ri.de (L. Graening).

[1] In the following, "design" refers to the 3D shape or topology of an engineering object, e.g., a car, and to the parameterization of the shape, e.g., a B-spline representation.

[2] In the following, "design quality" refers to one or several criteria that evaluate the performance of a design, e.g. based on the results of CAE simulations.

http://dx.doi.org/10.1016/j.aei.2014.03.002
1474-0346/© 2014 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-SA license (http://creativecommons.org/licenses/by-nc-sa/3.0/).

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

for an efficient design process. On the other hand, we can expect that new important insight about the design can be gained only when we examine the data holistically and relate previously unrelated parts of the design process to each other.

In a more formalized way, the targeted approach is illustrated in Fig. 1. The engineering design process is considered to be best described as a goal oriented iterative decision making process [4]. In each iteration, engineers decide about individual or a sequence of design variations, which lead to a final design configuration fulfilling pre-defined constraints and design goals best. The overall design process is spatiotemporally decomposed into a number of (multi-disciplinary) sub-processes $\{1,\ldots,i,j,\ldots,P\}$. Based on the result of a decision making process (DMP), each sub-process defines design changes that contribute to the synthesis process (SP) for the finally submitted design. The aim of the paper is to propose an approach that allows cross-process design data management (DB) and that enables the analytics process (AP) to integrate knowledge and information gained from all sub-processes. Finally, the results of the holistic analysis can be fed back to the individual sub-processes to improve the individual decision making.

Apart from the problem of relating different design representations to each other in the overall design process, in general the application of data mining techniques to engineering data has been less explored than, e.g. to economic data. Literature related to the extraction of human readable knowledge in the field of aerodynamic and structural design is rare. The team of Obayashi [5,6] have addressed the extraction of knowledge from a given data set in order to gain insights into the relationship between geometry and multi-criteria performance measurements. The authors applied self-organizing maps (SOM) in order to find groups of similar designs for multi-criteria performance improvements and trade-offs, and used the analysis of variance technique (ANOVA) to identify the most important design parameters. Their methods have been applied to supersonic wing design. In [7] the use of methods from information theory have been studied to reveal higher order interrelations between design and flow field properties. Their methods have been tested in the domain of turbine blade and passenger car design.

In most of the literature, the extracted information is linked to a specific and well-defined representation being used in the design process. Thus, the usability of the extracted information beyond this particular design and optimization process is only possible to a limited extent. Therefore, Graening et al. in [8] started to study the use of data mining techniques on a unified object representation. However, data mining based on such a typically high dimensional representation goes beyond the application of individual modeling technologies. Furthermore, it requires the consideration of other data mining aspects like, feature extraction, feature reduction and post-processing. Wilkinson et al. [9] adopted the basic idea of Graening et al. and utilized unstructured surface meshes as unified object representation for the prediction of the local wind pressure distribution on tall buildings.

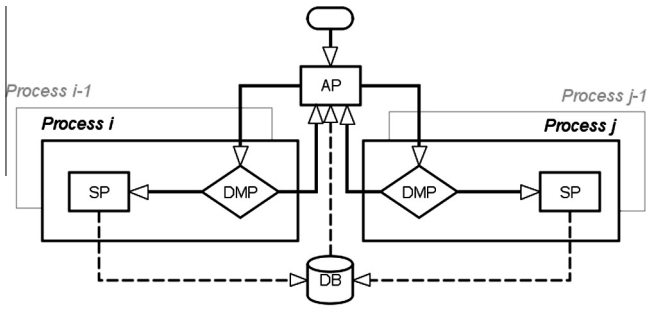



**Fig. 1.** A formalized view on the design process including the design synthesis (SP), the decision making (DMP) and the analysis (AP) process, see the text for a detailed explanation.

In this paper, we generalize the concept behind the analytics of design data based on a unified shape representation by introducing the shape mining framework. The remainder of the paper is organized in two parts. In the first part, we discuss different methods for shape mining and embed them into an overall framework. In the second part, the shape mining framework is applied to the analysis of passenger car design data.

The first part is divided into four sections. In Section 2, a unified design representation[3] is defined together with methods for the evaluation of local design differences. In Sections 3–5, methods for sensitivity analysis, for the extraction of design concepts, and for interaction analysis are introduced and discussed.

The second part of the paper is organized almost synonymously with the first part. Firstly, elements of different design processes that are the sources for the passenger car design data are described in Section 6. Statistical methods are applied to the meta design representation in Section 7, e.g. to investigate the course of design processes. In Sections 8–10 the methods from part one for sensitivity analysis, the extraction of design concepts, and the interaction analysis are applied to the data from the industrial design process. The aim is to model and understand the relation between shape variations of the car and changes in their aerodynamic quality.

Whereas part two of the paper is an application specific example, the approach presented in part one is generally applicable to all problems in the area of shape or topology mining. At the same time, some readers might find it useful to see the practical use of algorithms introduced in part one immediately; those readers are invited to read, e.g., Section 8 after Section 3. The paper closes in Section 11 with a conclusion and summary of the work.

## Part I: shape mining

More recently, technologies from computational intelligence and data mining, e.g., see [5,6], have been adopted to exploit experimental design data and computational resources for the support of engineers in the decision making process. However, the multi-disciplinary characteristics of complex design processes and the huge variability in computational design representations hinders the analysis of design data beyond individual design configurations and processes. Especially the variation in the computational representations being used makes an efficient knowledge exchange between design processes difficult.

The shape mining framework, as illustrated in Fig. 2, targets the integration of technologies for the implementation of a holistic analysis processes. It requires the transformation of designs into a meta-representation, which facilitates the evaluation of design differences on a holistic basis. Just the transformation of the designs into such a unified meta-representation, together with the evaluation of design quality differences, allows a holistic modeling of the design data independently of the originating process. Depending on the stated problem, modeling techniques from data mining and machine learning are applicable to investigate design sensitivities, retrieve abstract design concepts and analyze the interrelations between distinct design parts with the focus to understand the interplay between local design differences and changes in their quality. The resulting knowledge from the analysis of the design data can be utilized to support engineers in decision making and to improve future design and optimization processes.

[3] In the following, we will use the terms "unified design representation" and "meta design representation" synonymously.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

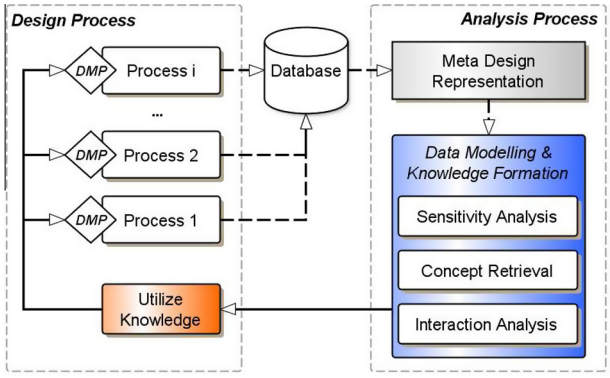



**Fig. 2.** High level view on the shape mining framework.

## 2. Meta design representation

For the analysis of three dimensional shapes in the domain of aerodynamic and structural design, unstructured surface meshes are adopted to build a unified representation.

### 2.1. Unstructured surface mesh

Unstructured surface meshes constitute a discrete geometrical representation of continuous object boundaries. Based on the terminology of Spanier [10] and Alexa [11], an unstructured surface mesh representation is defined as follows:

**Unstructured surface mesh**: An unstructured polygonal surface mesh $\mathcal{M}$ is a piecewise linear approximation of the boundary of an object. Each surface mesh $\mathcal{M}$ is defined by a pair $\mathcal{M}:(\mathcal{V},\mathcal{K})$, where $\mathcal{V}$ is a set of vertices $\mathcal{V}=(\vec{v}_1,\ldots,\vec{v}_n)$, with $\vec{v}_i\in\mathbb{R}^m$, defining the geometric position of $n$ points sampled from the continuous design boundary. The complex $\mathcal{K}$ is a set of $p$ simplices of the form $\{i_1,i_2,i_3,\ldots,i_\mu\}$, with $i_l, l\in[1\ldots n]$ defining a set of vertices that enclose a polygonal face made up of $\mu$ segments.

Given a continuous surface $\mathcal{S}$ the list of vertices $\mathcal{V}$ form a finite set of surface points in the Euclidean space with $\mathcal{V}\subseteq\mathcal{S}$. The polygonal faces, defined by the simplices $p\in\mathcal{K}$, make up a list of surface patches building a local linear approximation of $\mathcal{S}$. Derived from the normal vectors of the surface patches, a list of normal vectors $\mathcal{N}=(\vec{n}_1,\ldots,\vec{n}_n),\vec{n}_i\in\mathbb{R}^3$ can be derived for each vertex, where the normal vector $\vec{n}_i$ has a defined direction perpendicular to the surface and provides local gradient information at the position of the vertex $\vec{v}_i$.

The simplicity of discrete unstructured surface meshes and the fact that nearly all 3D objects can be transferred into such a geometrical representation makes them an adequate choice for the exchange of shape information and knowledge between design processes and engineers.

### 2.2. Local surface differences

Local features of the surface mesh can be derived to describe surface properties at the location of individual vertices or surface patches, e.g. related to the absolute vertex position or to the curvature of the surface, e.g. see [12] for a comprehensive overview on geometrical shape properties. Modifications of a design can lead to variations in the surface features, which can be the cause for changes of the functional properties of a design. The definition of the *local surface difference* targets to provide a general means to formalize and quantify those design variations.

**Local surface difference**: Given two surface mesh representations, a reference mesh $\mathcal{M}_r$ and a modified mesh $\mathcal{M}_m$, together with a finite set of vertices $\mathcal{V}_r$ and $\mathcal{V}_m$ respectively, for each vertex $i$ with $\vec{v}_i^r$ the local surface difference $\Delta_i^r=D(f_i^r,f_j^m)$ is defined as the difference between the feature $f_i^r$ assigned to $v_i^r$ and the feature $f_j^m$ linked to its corresponding vertex $\vec{v}_j^m$.

The calculation of the local surface differences involves the quantification of surface features at the position of each vertex, the identification of corresponding vertices and the quantification of the related feature difference. Both, the determination of the kind of features that is calculated as well as the strategy used for identifying corresponding vertices needs to be tuned to the class of applications and the kind of modifications under consideration.

*Vertex displacement.* Exploiting information about the absolute position of the vertices, the measure of displacement implements an effective way to quantify local surface differences relative to a chosen reference mesh $\mathcal{M}_r$. The displacement is formally defined as:

$$\Delta_i^r\triangleq\delta_{i,j}^{r,m}=\delta(\vec{v}_i^r,\vec{v}_j^m)=(\vec{v}_j^m-\vec{v}_i^r)\circ\vec{n}_i^r,\delta\in(-\infty,+\infty), \tag{1}$$

where $\vec{v}_i^r$ defines vertex $i$ of mesh $\mathcal{M}_r$, $\vec{v}_j^m$ the corresponding vertex $j$ of mesh $\mathcal{M}_m$, $\vec{n}_i^r$ the normal vector assigned to vertex $i$ and $\circ$ the scalar product. As such, the displacement $\delta_{i,j}^{r,m}$ measures the vertex difference in the direction of the vertex normal relative to $\mathcal{M}_r$. The magnitude of $\delta_{i,j}^{r,m}$ provides information on the amount and the sign information about the direction of vertex modifications. Details related to the properties of the displacement quantity are studied by Graening et al. [8] in more detail. Depending on the number of designs and the number of vertices, the evaluation of the displacements can become computational expensive. Therefore, a fast approximation based on already calculated displacement data is suggested in [13].

### 2.3. Identification of corresponding vertices

An appropriate identification of corresponding vertices is essential to measure the correct feature difference between vertices of two different surface meshes. Wrong estimates of correspondence will lead to an error in the measurements and hence to errors in any subsequent data analysis step. Let $\mathcal{M}_r:(\mathcal{V}_r,\mathcal{K}_r)$ and $\mathcal{M}_m:(\mathcal{V}_m,\mathcal{K}_m)$ be two unstructured surface meshes, where $\mathcal{M}_r$ defines a reference and $\mathcal{M}_m$ a modified mesh. The main objective in solving the correspondence problem is to find an appropriate function $f$ which maps each vertex $\vec{v}_i^r\in\mathcal{V}_r$ to a corresponding vertex $\vec{v}_j^m\in\mathcal{V}_m, f:\mathcal{V}_r\to\mathcal{V}_m$.

A global solution to the correspondence problem, which implements a universal transfer function $f$ that results in an "exact" mapping, does not exist. A correct identification of the corresponding points is only possible if the transformation that maps $\mathcal{M}_r$ to $\mathcal{M}_m$ is known. Constrained by the diversity of possible design variations, a specific mapping function or algorithm has to be chosen. In this work, the assumption is made that no structural design modifications, which change the global characteristics of the design are applied.

Taking only the relative position of the vertices into account, the Euclidean distance between vertices can be used to find a solution for $f$, with

$$f(\vec{v}_i^r)=\min_{j\in n_m}\left\{|\vec{v}_i^r-\vec{v}_j^m|\right\}, \tag{2}$$

where $n_m$ equals the number of vertices that make up the surface mesh of the target design. In general, it is possible that one vertex from mesh $\mathcal{M}_m$ is assigned to more than one vertex of mesh $\mathcal{M}_r$.

The identification of corresponding vertices using the Euclidean distance has weaknesses in areas where edges or solid structures are deformed. For such cases, including curvature information into $f$ can overcome otherwise false matches. Therefore, the following functional implements an extented matching strategy:

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

$$f_n(\vec{v}_i^r) = \min_{j \in n_m} \left\{ |\vec{v}_i^r - \vec{v}_j^m| \cdot (2 - \vec{n}_i^r \circ \vec{n}_j^m) \right\}, \quad (3)$$

where $\vec{n}_i^r \circ \vec{n}_j^m$ quantifies the difference between the normal vectors of two vertices. The objective function has been defined so that the vertex $\vec{v}_j^m$ that is closest to $\vec{v}_i^r$ and has a similar normal vector direction is assigned as the corresponding vertex. The objective function is a simplification of the one used by Wang [14]. If affine transformations, like rotation, scaling or translation are applied to the design, refined algorithms as suggested by McKay [15,16] can be applied.

## 3. Sensitivity analysis

One of the main challenges in decision making within an efficient design process is to be able to predict the effect of design variations on the design quality. Considering $\Delta_i^r$ as a measure of design variations and $\phi^r$ as a measure of the quality difference between two designs (estimated with respect to a particular reference design $r$), sensitivity analysis targets to reveal the influence of $\Delta_i^r$ on the quality change $\phi^r$, e.g. reflecting the difference in the aerodynamic quality of designs. In other words, design sensitivity analysis is the study and estimation of the impact of design feature variations on the variation in the overall design quality. In the process of shape mining, the design sensitivity analysis is carried out for two purposes, the extraction of knowledge about the importance of particular design areas with respect to the target variable by means of sensitivity estimation, as well as the ranking and filtering of design features to improve subsequent modeling and data mining steps by means of identifying sensitive design areas. A more general introduction into the field of sensitivity analysis is given by Saltelli et al. [17].

### 3.1. Sensitivity estimation

A wide range of methods for the quantification of parameter sensitivity have been studied. *Local sensitivity analysis* aims at directly estimating the gradient at a certain fixed point in the design space. This is related to adjoint modeling [18,19] and automatic differentiation [20]. Following Saltelli et al. [17], local methods are only informative at the position of one design point. In contrast to local methods, *global sensitivity analysis* aims at estimating the relation between input and output variables, given a limited number of data samples over a larger design or input space. Ascough et al. [21] provide a qualitative evaluation of the most popular methods. Among them are the Fourier Amplitude Sensitivity Test (FAST) [22], the Sobol' method [23] and the mutual information index (MII) [24]. For the estimation of the total correlation, the FAST and Sobol' method apply a multi-factor analysis taking the interrelation and covariances between all independent variables into account. Although, multi-variate correlation analysis allows a more correct quantification of the sensitivities, for a large number of variables its applicability is limited due to the high computational expense. Univariate correlation analysis provides only a qualitative estimate of the importance of the variables but with manageable computational costs. It should be noted that using univariate correlation analysis, co-variations regarding other factors, which are not included in the analysis, act as structural noise and can bias the sensitivity estimates.

The most prominent univariate correlation measures are the Pearson and Spearman correlation coefficients. Given $\Delta_i^r$ as a measure of local design feature variations and $\phi^r$ as the variation in a particular design quality, the Pearson or Spearman correlation coefficient $r_{\Delta_i}^r$ estimates the linear correlation between surface variations and quality changes, either with respect to the measured parameter values or their ranks, see [8] for further details. As such, the correlation coefficient provides information about an expected increase or decrease in the design quality with respect to an increase or decrease in the local shape feature relative to a chosen reference design $r$.

Applying the Pearson or Spearman correlation coefficient one makes strong assumptions on the linearity of the relation either between the feature values or their ranks. In contrast, information theoretic formulations like mutual information quantify the association between variables by comparing the underlying probability distributions, and as such making no assumptions on the nature of the interrelation. Rather, the information theoretic approach investigates the co-occurrence of feature and quality values. When applying mutual information to the sensitivity analysis [24], the value of the mutual information is typically normalized by the entropy of the dependent variable by means of the design quality index. In engineering, the number of available designs is often limited due to the high expense in the design quality evaluation, which can make a correct estimation of the probability distributions difficult. To improve the estimation of the mutual information for real-valued design variables, Graening et al. [25] have investigated a robust variant of the mutual information, which has shown to provide more reliable sensitivity estimates when analyzing a low number of designs.

Independent of the method, all sensitivity estimates are carried out at the position of the reference design. The remaining designs and their related feature variations considered define the scope of the sensitivity analysis. Fukushige et al. [13] have used a K-Nearest Neighbor approach to restrict the sensitivity estimation to a local vicinity around a chosen reference design. Such improvements of the sensitivity estimation become relevant if the number of designs in the database is huge and if a large area of the entire design space has been sampled.

### 3.2. Identification of sensitive areas

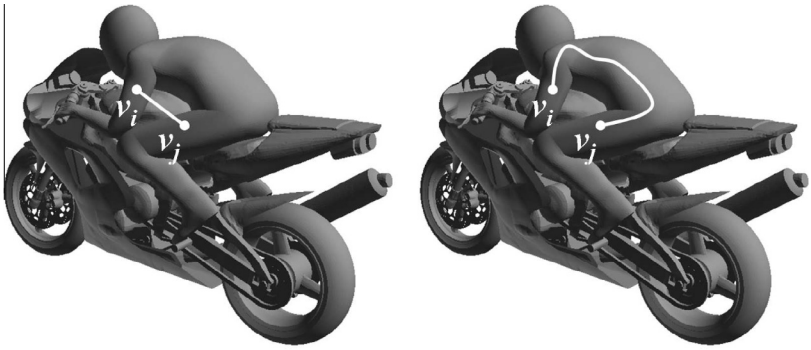
Applying enhanced modeling techniques from data mining or artificial intelligence to analyze the interrelations between surface feature and design quality changes, which goes beyond univariate sensitivity analysis, is getting impractical due the high dimensionality of surface mesh representation. Typically, the number of vertices that build the discrete surface mesh is large with $n \gg 1000$ in order to model all possible design changes. With the target to model higher order relationships between distant design variations and design qualities, a low dimensional manifold in the input space is favored. Given the results of the sensitivity analysis and a measure of the proximity of vertices on the surface, we introduce an explicit feature reduction step, which is tuned to the unified meta representation. Thereafter, nearby sensitive vertices are grouped into so called sensitive design areas:

**Sensitive design area**: A sensitive design area $\mathcal{A}$ is defined as a subset of vertices $\mathcal{A} = \{\vec{v}_1, \ldots, \vec{v}_l\}$ of the entire mesh $\mathcal{M}$, which form a closed area on the surface and are similar in their sensitivity to the considered design quality.

Given a pre-defined distance measure $d_{i,j} \mapsto f(d_{i,j}(\mathcal{M}), d_{i,j}(\mathcal{R}))$, vertices $\vec{v}_i$ and $\vec{v}_j$ of mesh $\mathcal{M}$ belong to one and the same design area if they are close with respect to spatial distance $d_{i,j}(\mathcal{M})$ and if they share a similar sensitivity $d_{i,j}(\mathcal{R})$, where $\mathcal{R} = \{r_1, \ldots, r_n\}$ contains the sensitivity estimates for all of the $n$ vertices of mesh $\mathcal{M}$.

*Distance measure.* The choice of the spatial distance measure is crucial for the retrieval of sensible design areas. Using the Euclidean distance [8] works well for simple shapes. However, it might underestimate the distance between vertices on the surface around areas of high curvature, see Fig. 3. Vertices which are close in the three dimensional space do not have to be close along the two dimensional manifold of the surface. As an alternative the geodesic

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002




(a) Euclidean distance (b) Geodesic distance

**Fig. 3.** Illustration of the conceptual difference between spatial distances.

distance is suggested, which measures the distance between vertices along the surface. It provides a better suited estimation of the proximity of vertices with respect to surface meshes. The geodesic distance exploits the connectivity within the triangulated mesh. Pairwise geodesic distances are calculated by identifying the shortest path between two points in the mesh using the algorithm of Dijkstra [26], and summing up the length of the edges along the identified path.

In addition to the spatial distance, $d_{ij}(\mathcal{R})$ is defined as the difference between the normalized sensitivity values $r_i$ and $r_j$. Finally, the overall similarity measure is defined as follows:

$$d_{i,j} = d_{i,j}(\mathcal{M}) \cdot (1 + d_{i,j}(\mathcal{S})), \tag{4}$$

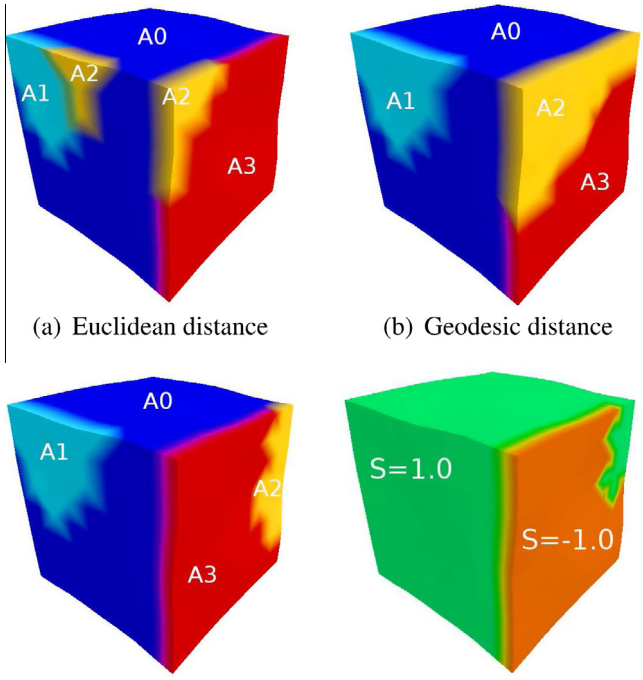
where $d_{i,j}(\mathcal{M})$ is either the Euclidean or geodesic distance.

*Automatic identification of sensitive areas* Based on the definition of sensitive design areas clustering techniques are applicable to automatically derive sensitive areas in an unsupervised manner, see Jain et al. [27] for a comprehensive overview of clustering techniques. Adapting the typical clustering procedure, the following steps constitute the automatic procedure for identifying sensitive areas:

1. Definition of the similarity measure (including spatial and sensitivity information).
2. Pre-select vertices based on their sensitivity (optional).
3. Select and apply the unsupervised clustering procedure to derive sensitive areas.
4. Determine the cluster centers.
5. Assign the local surface differences of the cluster centers and the design related properties to a reduced data set.

Before carrying out the grouping of vertices, non-sensitive vertices can be removed from the set either based on statistical significance tests or on a simple threshold calculation. After that, a standard clustering method, like the K-Means algorithm can be adopted to partition the surface into distinct surface areas. Using more sophisticated clustering techniques like X-Means [28] or the gap statistic [29] can overcome the problem that the number of clusters needs to be defined in advance.

*Example.* Fig. 4 illustrates an artificial example, outlining the basic idea for the identification of sensitive areas, where vertices are automatically grouped based on a pre-defined similarity measure. In the example, the reference design is defined by the triangulated surface mesh of a simple cube. An arbitrary set of vertices has been selected from the mesh for clustering. In the illustration, vertices which belong to the dark blue area, labeled with $\mathcal{A}_0$ (A0), are not considered for clustering. Fig. 4(a) and (b) shows the clustering results using the Euclidean and the geodesic distance as similarity




(a) Euclidean distance (b) Geodesic distance

(c) Geodesic distance & sensitivity difference

**Fig. 4.** Artificial example: vertices are grouped based on different similarity measures.

measure, respectively. K-means clustering, with $k = 3$, has been applied for grouping the vertices on the surface using either of the two spatial distances. It can be seen that using the geodesic distance results in distinct closed areas $\mathcal{A}_1$ to $\mathcal{A}_3$, while using the Euclidean distance results in an unfavorable separation of area $\mathcal{A}_2$.

Next, sensitivity values are assigned to the vertices. As depicted in Fig. 4(c) on the right, artificial sensitivity values of $S = 1.0$ (green[4]) and $S = -1.0$ (orange) are assigned to the vertices. Using the distance measure from Eq. (4), which is defined by the superposition of the sensitivity information and the geodesic distance, results in a different grouping shown in Fig. 4(c) on the left. As desired, the K-Means clustering algorithm groups nearby vertices with the same sensitivity value into the same group of vertices, see $\mathcal{A}_2$ and $\mathcal{A}_3$ in Fig. 4(c) in comparison to (c). Finally, the clustering procedure results in three sensitive areas along the cube surface.

[4] For interpretation of color in the different figures, the reader is referred to the web version of this article.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

### 3.3. Utilizing sensitivity information

The sensitivity analysis can provide engineers with new insights into the cause of changes in functional design properties. It allows engineers to predict the function of a design based on planned shape variations. In Graening et al. [8] the validity of the calculated sensitivities has been tested given the data of turbine blade geometries. Direct manipulation of free form deformation (DMFFD) [30] has been applied to create concrete shape deformations for the comparison of the predicted and the evaluated aerodynamic properties of the designs. Sensitivity information can be utilized to construct an initial object representation for subsequent computational optimizations. Using free-form deformation techniques, Graening et al. [25] have shown that an adaptation of the initial representation based on sensitivity information can lead to an improved optimization process.

## 4. Design concepts

Feature reduction by the identification of sensitive areas facilitates an enhanced modeling of the design data. In this section, particular techniques from data mining and machine learning are investigated to derive, describe and evaluate abstract design concepts, which are defined as follows:

**Design Concept** A design concept is an abstract representation of a class of designs sharing an akin characteristic that map to approximately equivalent design qualities.

Generalizing from individual design solutions, the identification and representation of design concepts can lead to a structuring of the design domain, based on which design concepts can be used to classify designs with respect to shape and quality. The algorithmic identification of concepts does support engineers in processing large amounts of design data. For example, it can help engineers to reveal common properties of a group of designs in order to patent real design properties or to evaluate newly discovered concepts against existing patents.

### 4.1. Concept representation

Concepts are represented by boundaries with respect to shape and quality variables that allows to discriminate designs of a common characteristic from other designs. A compact and human readable representation of the boundaries is desired to ease a subsequent interpretation and processing of the concepts by the engineer. In the field of machine learning and data mining, IF–Then rules are often used to represent such abstractions in human-readable form, which are formally defined as:

*IF* [antecedent] *THEN* [consequent],

or using a more compact notation {antecedent} → {consequent}, or $A \rightarrow C$, e.g. see [31] for an introduction into the topics of rule induction and association rule learning. When adopting the formulation of rules for the description of design concepts, the antecedent $A$ represents an abstract object specification, e.g. defining the object shape, and the consequent $C$ defines design quality related properties. Depending on the level of abstraction and the nature of the data, rules can be categorized into qualitative and quantitative rules. Qualitative rules refer to discrete states of the related variables, while quantitative rules [32] consider the variables explicitly as real-valued attributes. For quantitative rules, the antecedent and consequent of a rule are composed of attribute interval relations, such that the attribute values of a design have to be within a defined range to be covered by the design concept. As an example the lower and upper bound of the intervals could be chosen as the standard deviation around the mean or as the minimum and maximum feature value of the designs covered by the design concept.

### 4.2. Concept evaluation

The computational identification of design concepts can result in a large set of concepts, which can hardly be handled by engineers or data mining experts. An a priori evaluation of design concepts allows to order and filter concepts based on their relevance, so that the most relevant concepts can be studied first and irrelevant concepts can be omitted. In the following, selected measures for the evaluation of design concept relevance are briefly reviewed based on [33,34] and a new measure of *utility* is introduced.

*Review of Existing Relevance Measures.* Measures of relevance or interestingness can be classified into *objective* and *subjective* measures. While objective measures like specificity or accuracy rely on the statistics of the raw data, subjective measures like interestingness and surprisingness take additionally engineer's experiences and preferences into account.

Given a concept representation of the form $A \rightarrow C$ and a data set $\boldsymbol{D}$, which covers $N$ design variations and quality values, the basic measures of relevance, namely coverage and support are defined as $\mathrm{cov}(A \rightarrow C, \boldsymbol{D}) = P(A)$ and $\mathrm{supp}(A \rightarrow C, \boldsymbol{D}) = P(AC)$ respectively. Given $D$, the probabilities $P(A)$ and $P(AC)$ quantify the likelihood that a design meets all conditions in $A$ or in $A$ and $C$, respectively. The support provides information about the generality of a design concept. Concepts with low support are often too specific to be of major relevance and concepts with very high support are likely to represent trivial associations which are already known to experts. If the confidence is high that the ascribed design specification, defined by $A$, causes the abstracted consequence $C$ an association is called reliable. In terms of support and coverage, the measure of confidence is defined as follows:

$$\mathrm{conf}(A \rightarrow C, \boldsymbol{D}) = \frac{\mathrm{supp}(A \rightarrow C, \boldsymbol{D})}{\mathrm{cov}(A \rightarrow C, \boldsymbol{D})} = P(C|A) = \frac{P(AC)}{P(A)}. \tag{5}$$

The confidence is low, if the concept covers many designs but has a low support. Design concepts with low confidence are potentially irrelevant to the engineer and can often be rejected from further considerations. However, using the confidence measure only to quantify the relevance of an association can result in misleading evaluations, see [35]. Under the assumption that the strength of the correlation between antecedent and consequences reflects the interestingness of an association, Tan and Kumar [35] introduced a new metric that takes the confidence of the association and its reversal into account:

$$\mathrm{IS}(A \rightarrow C, \boldsymbol{D}) = \sqrt{\mathrm{conf}(A \rightarrow C, \boldsymbol{D}) \times \mathrm{conf}(C \rightarrow A, \boldsymbol{D})}. \tag{6}$$

*IS* has been shown to allow a sensible ordering of the associations according to the interestingness assumption made.

*Measure of Utility.* The support, confidence as well as the interestingness measure alone do often not provide an adequate measure to quantify the relevance of an association. In existing evaluation methods the expected quality of a concept or association and the objectives of the design process are not considered. In the engineering domain, objective values that quantify the design goal are typically well defined, allowing us to derive a measures of relevance quantifying the utility of a design concept.

Design engineers typically define a set of objective values $O = \{o_1, \ldots, o_n\}$, e.g. targeting the minimization of all the objective values, $\min(o_1, \ldots, o_n)$. For example, objective values can relate to the minimization of manufacturing costs, the minimization of fuel consumption or the maximization of the car volume. In the context of shape optimization, objective values are typically formulated based on the design quality as well as on the shape itself. In the

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

here defined concept representation, $A$ refers to the shape definition and $C$ to the related design quality, so that $A \wedge C \rightarrow O$. For each concept, $O$ specifies the expected objective values given the specification of $A$ and $C$.

In the case that $O$ contains multiple quantitative attributes, one can adopt performance values used in multi-objective optimization algorithms [36] to evaluate the compliance of a concept with the objectives. One established measure is the hypervolume indicator, see [37]. Following While et al.: "The hypervolume of a set of solutions measures the size of the portion of objective space that is dominated by those solutions collectively.", where a solution $a$ is said to dominate a solution $b$ if for each objective, solution $a$ equals or outperforms solution $b$, and solution $a$ at least outperforms $b$ with respect to one objective. Thus, based on a set of design solutions $\boldsymbol{D}$ the compliance of a concept with the objectives $O$ can be estimated by calculating the hypervolume $\mathrm{vol}(O, \boldsymbol{D})$ based on the non-dominated solutions of all designs covered by the concept.

Combining the hypervolume indicator and the *IS* metric, a new measure of relevance is defined to evaluate the utility of a design concept:

$$\mathrm{util}(A \rightarrow C, \boldsymbol{D}) = \mathrm{vol}(O, \boldsymbol{D}) \cdot \mathrm{IS}(A \rightarrow C, \boldsymbol{D}). \tag{7}$$

Thereafter, the utility measure is defined as the product of the hypervolume $\mathrm{vol}(O, \boldsymbol{D})$ and the *IS* metric. According to the *IS* metric, defined in Eq. 6, a design concept is of high utility if it has a high confidence that the association $(A \rightarrow C)$ described by the concept is true for all designs in $\boldsymbol{D}$. Furthermore, a concept is of high utility, if according to the calculation of $\mathrm{vol}(O, \boldsymbol{D})$, the related design changes are expected to result in a high design quality, by means of complying with the pre-defined design objectives.

*Example.* A simple illustrative example should clarify the differences between the different measures of relevance. In the example, three design concepts **a**, **b** and **c** are defined, as depicted in form of dashed rectangles in Fig. 5. The concepts are evaluated against a data set $\boldsymbol{D}$ containing $N = 14$ solutions. Fig. 5 plots all $N = 14$ solutions and its link to the design concepts in the design Fig. 5(a) and the objective space Fig. 5(b). The variables $\Delta_1$ and $\Delta_2$ refer to design variables quantifying the surface difference, e.g., the displacement between corresponding vertices, see Section 2.2. The objective values $O$ are defined based on the design quality differences $\phi_1$ and $\phi_2$, with the goal to minimize both, $\phi_1$ and $\phi_2$ simultaneously.

Each of the concepts **a**, **b** and **c** can be transferred into a human readable form, as described in Section 4.1. For example, concept **a** can be written as follows:

$$a: \ \Delta_1 \in [0.0, 0.4] \wedge \Delta_2 \in [0.6, 1.0] \rightarrow \phi_1 \in [0.0, 0.4] \wedge \phi_2 \in [0.6, 0.9].$$

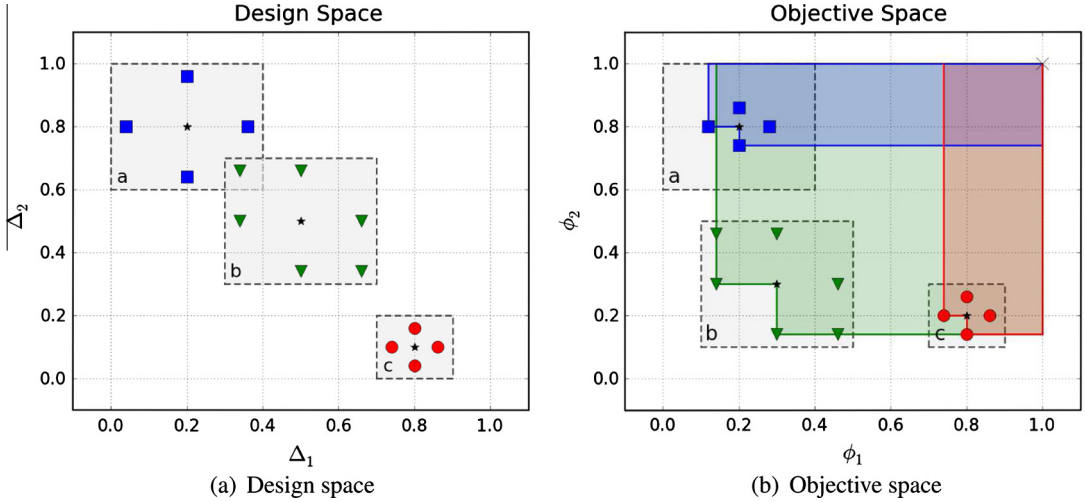
The example rule implies that if the values of the design variables $\Delta_1$ and $\Delta_2$ lie within the given interval, then the objective values $\phi_1$ and $\phi_2$ will also lie in the specified intervals. Therefore, from rule **a** it can be expected that a joint modification of the surface related to vertices $\vec{v}_1$ and $\vec{v}_2$, with $\Delta_1 \in [0.0, 0.4]$ and $\Delta_2 \in [0.6, 1.0]$, results in a change of the design qualities $\phi_1$ and $\phi_2$ in the range of $\phi_1 \in [0.0, 0.4]$ and $\phi_2 \in [0.6, 0.9]$, respectively.

According to Section 4.2, for each concept **a**, **b** and **c** the coverage, support, confidence and *IS* measure have been calculated based on the probabilities $P(A), P(C)$ and $P(A, C)$, estimated by the respective relative frequencies from all data in data set $\boldsymbol{D}$. In order to evaluate the utility (Eq. 7), the expected hypervolume $\mathrm{vol}(O, \boldsymbol{D})$ for each concept is calculated with respect to the reference point $(\phi_1, \phi_2)^T = (1.0, 1.0)^T$, as illustrated in Fig. 5(b). The reference point defines the worst assumable design quality. The results of the evaluations are summarized in Table 1, ordered by the measure of utility.

Concept **b** covers the largest proportion of the design solutions, followed by **a** and **c**. Concepts **b** and **c** show that the maximum confidence value $\mathrm{conf}(A \rightarrow C, \boldsymbol{D}) = 1.0$ is reached if the support equals the coverage value. $IS = 1.0$ only if the confidence of the reversal of the association $\mathrm{conf}(C \rightarrow A, \boldsymbol{D}) = 1.0$ as well. If *IS* would be applied to the evaluation of relevance, concepts **b** and **c** would be equally ranked. However, considering that both $\phi_1$ and $\phi_2$ should be minimized, engineers would clearly favor concept **b** over concept **c**. This is reflected by the introduced measure of utility.

### 4.3. Design concept learning

The automatic retrieval of potential design concepts from a set of designs is directly related to concept learning [38] and classification algorithms [39]. In machine learning and data mining a large amount of classification algorithms have been studied that differ mostly in the way the classification boundaries are constructed. Among the most prominent ones are Decision Trees [40], Rough Set Theory [41], Fuzzy Sets [42], Artificial Neural Networks [43] and Support Vector Machines [44]. The choice of the classifier should be based on the characteristics of the design data, the




(a) Design space (b) Objective space

**Fig. 5.** Example design data set, where solutions are grouped into three different concepts a, b and c.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

**Table 1**
Evaluation results of the concept candidates (Fig. 5) ordered according to their utility value.

| Concept | cov | supp | conf | IS | vol | util |
|---|---|---|---|---|---|---|
| b | 0.429 | 0.429 | 1.000 | 1.000 | 0.714 | **0.714** |
| c | 0.286 | 0.286 | 1.000 | 1.000 | 0.220 | **0.220** |
| a | 0.357 | 0.286 | 0.801 | 0.895 | 0.224 | **0.200** |

classification error as well as on the possibilities to represent the retrieved concepts in a human readable way. Two classification algorithms, the decision trees and the self-organizing map are briefly introduced.

*Decision Trees.* Decision trees are supervised learning models frequently used in data mining, machine learning and other domains. Their popularity comes from their conceptual simplicity, from their interpretable structure and because they can be applied to regression and classification tasks similarly [40,45,46]. Decision trees are constructed by recursively splitting the input space into hyper-rectangular sub-spaces. They are represented by a directed graph that consists of a finite set of nodes and branches connecting them. One distinguishes between the root node, internal or test nodes and terminal nodes representing the leaves of the tree. The root and internal nodes represent attributes at which conditions are tested, splitting the solution set into two or more homogenous sub-sets. A class or target value is assigned to each node abstracting the characteristics of the designs in the represented sub-space. At each node the variable and split-point is chosen based on a quality measure, minimizing the impurity at each node, e.g., misclassification error, the gini index or the cross-entropy. Pruning strategies can be applied for a subsequent shrinkage of large decision trees. Each branch in the final tree can be transferred into an association rule by processing each node along the branch.

*Self Organizing Maps.* Motivated from cortical maps, Kohonen's Self-Organizing Maps (SOM) [47,48] belong to the class of unsupervised artificial neural networks. Meanwhile SOMs have been adopted to a broad variety of applications, e.g. see [49]. For the investigation of structured aerodynamic design data SOMs have first been studied by Obayashi et al. [50] in order to group and analyze the trade-off of aerodynamic designs with respect to multiple performance criteria. SOMs implement a feed-forward network structure consisting of two layers, one input and one output layer, referred to as the feature map. The structure of the input layer is directly defined by the number of input variables while the topology of the output layer needs to be pre-defined a priori to the SOM training procedure. Typically a 1D or 2D feature map is used where neurons are organized on a regular lattice. Each neuron of the input layer is fully connected to the neurons of the output layer by continuous weights. The training algorithm realizes a topology preserving mapping from the input space to the low dimensional feature map by iteratively applying competitive learning and cooperative updating to the adaptation of the weight vectors. After the training phase, each weight vector connected to each output neuron makes up a prototype vector, representing a class of similar input vectors. The low dimensional output layer preserves the statistics and structure of the input data set. The investigation of the output neurons thus unveils information about the structure of the high dimensional input data.

Although, the visualization of information of the output neurons can already provide a deep understanding of the organization of the input data, the correct interpretation of the results requires knowledge about the underlying SOM principles. The extraction of linguistic rules from the trained network can provide a direct access to the concepts and ease their interpretation. Based on a trained SOM, sophisticated rule extraction algorithms perform an additional abstraction and rule extraction step, see [51–53].

### 4.4. Utilizing information about design concepts

The extracted concepts and formulated rules can be directly utilized within a knowledge based engineering system, see [54] for an introduction, or to build up expert systems for distinct design problems. In combination with the universal design representation concepts linked to the holistic design can be formulated. As such the acquired design concepts and their formulation in linguistic form can directly help engineers in decision making, beyond individual processes. Depending on the overall strategy, sparsely sampled areas in the design space can be further explored, or the information about outperforming concepts can be exploited to guide the design process. The analysis can lead to the discovery of new design concepts and hypothesis which can be validated in subsequent experimental or simulation studies. Furthermore, the clear description of such concepts can reveal relevant interrelations between design parts, domains and engineers, upon which communication strategies can be revisited. In computational optimization, global search algorithms like evolutionary strategies typically employ strategy parameters to guide the search process. The a priori adaptation of the strategy parameters has a positive effect on the performance of the search algorithm as shown in [55]. The initialization of those strategy parameters prior to the optimization run, or the definition of optimization constraints based on the acquired knowledge could further increase its efficiency.

## 5. Interaction analysis

As stated in Section 1, complex design problems are often decomposed into smaller subsystems which are optimized in parallel. Finding a proper problem decomposition is not trivial and has a strong affect on the efficiency of the overall design process. In practice, the decomposition is mostly done based on engineers experiences and remains fixed over time. Interaction analysis targets an automatic identification and analysis of interrelated sub-components. It investigates the interplay between different components and the objectives defined by the engineers. As an example, the influence of a formula one car's rear wing on the overall downforce of the car strongly depends on the correct adjustment of the front wing angle. The identification and analysis of those interactions is an important step to understand and improve the overall system behavior.

Adopting the definition from Krippendorff [56], we define design interactions as follows:

**Design Interaction** A design interaction is defined as a unique dependency between design and objective parameters from which all dependencies of lower ordinality are removed.

Mostly, mathematical approaches for the quantification of interactions can either be classified into methods of variance decomposition like ANOVA or into probabilistic methods as applied in the field of information theory. In the following, *interaction information* as one of the most general attempts for evaluating parameter interactions is reviewed. Compared to methods of variance decomposition, the measure of interaction information can be equally applied to continuous, discrete and qualitative data.

### 5.1. Interaction information

Information theoretic attempts to quantify interactions are based on the Shannon entropy. The Shannon entropy for a discrete random variable $X_i$ is denoted as $H(X_i)$. For two variables $X_i$ and $X_j$, the mutual information $I(X_i;X_j)$ measures the amount of information shared between both variables, with:

$$I(X_i;X_j) = H(X_i) + H(X_j) - H(X_i,X_j), \tag{8}$$

where $H(X_i, X_j)$ is the entropy of the joint distribution of $X_i$ and $X_j$. The mutual information quantifies interactions of ordinality two, sometimes referred to as two-way interaction.

Based on the work of McGill [57], Jakulin [58] introduced the interaction information as an extension of the mutual information to multiple attributes $\mathcal{S} = \{X_i, \ldots, X_n\}$. The n-way interaction information $I(\mathcal{S})$ is defined as an iterating sum over marginal and joint entropies:

$$I(\mathcal{S}) = -\sum_{\mathcal{T} \subseteq \mathcal{S}} (-1)^{|\mathcal{S}|-|\mathcal{T}|} H(\mathcal{T}), \tag{9}$$

where $\mathcal{T}$ denotes any possible subset of $\mathcal{S}$. The formulation of Jakulin provides the theoretical basis for the quantification of interactions of arbitrary ordinality. For three variables $\mathcal{S} = \{X_i, X_j, X_k\}$ the interaction information assesses the amount of information that is unique to all three variables and is not given by any of the 2-way interactions. The three-way interaction in terms of mutual information can be formulated as follows:

$$I(X_i; X_j; X_k) = I(X_i, X_j; X_k) - I(X_i; X_k) - I(X_j; X_k), \tag{10}$$

where $I(X_i, X_j; X_k)$ defines the expected amount of information that $X_i$ and $X_j$ together convey about $X_k$. Since $I(X_i, X_j; X_k)$ is symmetric, this holds for any permutation of $i, j$ and $k$. In contrast to the mutual information, the interaction information can either be positive or negative. Jakulin [58] interpreted positive values of the interaction information as *synergy* and negative values as *redundancy*.

*Synergy:* The interaction information becomes positive if $I(X_i, X_j; X_k)$ is larger than the sum of the information that each variable $X_i$ and $X_j$ conveys about the third variable $X_k$. Thus, the synergy of $X_i$ and $X_j$ provides additional information about $X_k$. Lets consider an example where the acceleration force, the weight and the engine power of a car are considered as random variables, neglecting the knowledge of the related physical laws. Both, the weight and the engine power alone would provide certain information about the acceleration capabilities of a car. However, just the additional information about the interplay between weight and engine power would allow a correct prediction of the car's acceleration force. The weight and the engine power are in a synergy relation to the acceleration force, and the interaction information becomes positive.

*Redundancy:* In cases where the joint information $I(X_i, X_j; X_k)$ is smaller than the sum of $I(X_i; X_k)$ and $I(X_j; X_k)$, $X_i$ and $X_j$ share redundant information conveyed about $X_k$. In an example, where the acceleration, the engine power and the torque are considered as random variables, the torque and the engine power mostly provide the same information about the car's acceleration force. These parameters are redundant with respect to the acceleration. The interaction information becomes negative.

The interaction information can get close to zero either in the absence of information due to synergy and redundancy, or if the synergy effect and the redundancy cancel each other out, see [59] for more details.

### 5.2. Interaction graphs

The results of an interaction analysis can be visualized using so called interaction graphs, introduced by [60]. The graph denotes the interaction structure between variables, which, for example, are characterizing design properties. Jakulin distinguishes between supervised and unsupervised graphs. As an example, Fig. 6 depicts the structure of the supervised variant, which contains information about 2- and 3-way interactions relative to the uncertainty of an a priori chosen target variable $Y$, e.g. defining the quality of the designs. All information quantities are normalized by $H(Y)$, expressing the contribution of the parameters and their interactions in terms of portions of reduced uncertainty about $Y$. A label with

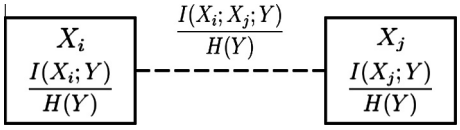



**Fig. 6.** Visualization of the information graph for 2- and 3-way interactions.

the value of the relative mutual information is assigned to the knots of the graph. The edges connecting two knots $X_i$ and $X_j$ are representing the values of the relative three-way interaction information $I(X_i; X_j; Y)/H(Y)$. The thickness of the edges is proportional to the absolute value of $I(X_i; X_j; Y)/H(Y)$ and the line style denotes its sign. Solid lines represent a positive (synergy) and dashed lines a negative interaction value (redundancy).

### 5.3. Utilizing interaction information

Applied to design parameters, the analysis of interactions provides a systematic and data driven approach to identify dependencies between design parts. On the one hand strongly interrelated design parameters need to be combined and optimized in one design process. Thus, the information about parameter interactions can be used to define the design representation. On the other hand the interaction analysis allows a decomposition of complex design problems into largely independent parts. In this context, the results of the interaction analysis can provide the means to reconsider the current implementation of the problem decompositions. Furthermore, results from the interaction analysis can also be used in model-based optimizations, e.g. to decide to represent the design space by two separate approximation models of a low degree of complexity.

## Part II: application to passenger car design

The basic concepts of the shape mining framework outlined in part I of the paper are applied to the analysis of passenger car design data in part II. The illustration in Fig. 7 provides an overview of the second part of the paper and relates the conceptual shape mining framework to individual application steps. Relating to the passenger car design process, the subsequent section deals with the representation, evaluation and design of the car shape for optimal aerodynamic performance. The designs that result from different design processes are transferred into unstructured surface meshes defining the unified representation that enables a holistic design data analysis, by means of shape mining. The resulting meta design data are the starting point for the extraction of

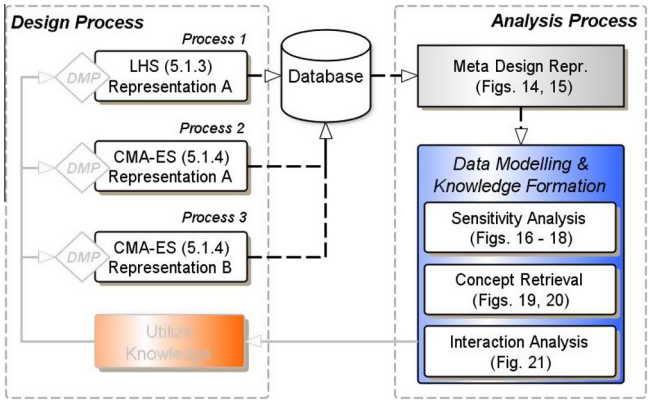



**Fig. 7.** High level view on the shape mining framework applied to the analysis of the passenger car design data.

relevant information from the design process. As summarized in Fig. 7, methodologies for sensitivity analysis, concept retrieval and interaction analysis are studied, e.g., to investigate the course of design processes, to identify the design areas which are sensitive to changes in the aerodynamic performance or to retrieve generalized car concepts.

## 6. Passenger car design synthesis

To underpin the practicality of concepts for knowledge extraction, a priori generated design data from a realistic application is needed. Typically, design data result from various diverse design processes where each design process follows a pre-defined strategy to reach a specific design goal. In this chapter two design strategies, as they are frequently used in CAE, are carried out to design the shape of a passenger car. The first one implements a global search strategy by means of uniformly sampling a constrained design space, while the second strategy follows a direct local search by exploiting the characteristics of the design during the progress of the design process. Both strategies result in design data sets, which are characteristic for explorative and exploitative design processes. Typically, a sensible combination of the two strategies is used, for both computational as well as human driven engineering design. The design space that represents all potential solutions is restricted by the representation of the passenger car. The improvement of the aerodynamic performance of the shape is pursued, with the overall design goal being formulated based on the results from computational fluid dynamic simulations (CFD).

### 6.1. Design representation

To model variations of the design of a passenger car Free Form Deformation (FFD) [61,62] has been applied to the initial car shape. FFD represents variations of a chosen baseline design, allowing global as well as local deformations, depending on the setup of the control grid relative to the embedded objects. Applying FFD to the passenger car shape requires the setup of a three dimensional control point grid. The control points of the grid serve as handles for the deformation of the embedded objects. The parameterized grid defines the degrees of freedom and constraints for the respective design processes. The choice of the representation strongly depends on the target setting. During the entire synthesis of a new design the representation seldom remains unchanged. As examples, two different control point grids have been constructed. The construction of the first one (representation A) incorporates expert knowledge about expected variations of the car shape, while the second control point grid (representation B) represents a standard set-up. For the definition of the control point grid and the deformation of the initial mesh an in-house software *VisControl* has been used. Finally, the variation of the control points results in a deformation of the embedded surface mesh representing the shape of the passenger car.

*Representation A.* The first control point grid of the FFD representation consists of $m_A = 567$ control points, $\mathcal{P}_A$. Splines of degree 3 and order 4 are utilized in the FFD representation. A significant portion of the control points has been introduced to constrain the deformations on the initial surface mesh $\mathcal{M}_I$, e.g. to limit deformations at the wheel house. Such limitations ensure, e.g. the manufacturability of the resulting car design. Based on the control volume, $k_A = 16$ control point groups, $\mathcal{G}_A = (CPG_1, \ldots, CPG_{k_A})^T$, have been defined. The individual control point groups are marked and labeled in Fig. 8(a)–(c). The displacement of the control points within each group is restricted to displacements along individual axes, e.g. control point groups $CPG_0$ to $CPG_3$ are restricted to modifications in $x$ direction. Variations in the $y$ direction are applied so

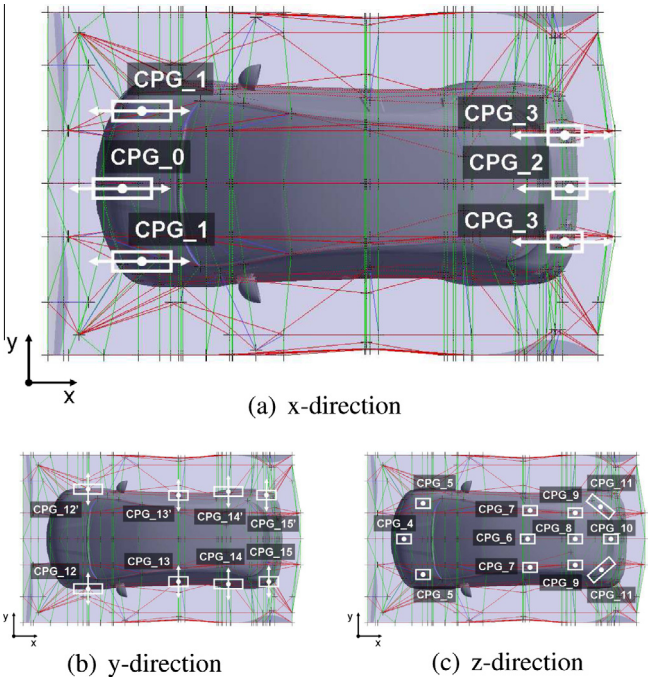



**Fig. 8.** Illustration of the 16 control point groups of **Representation A**.

that the symmetry of the car shape is kept. The introduced control point groups define $k_A = 16$ tunable parameters for defining new car shapes. Formally, given variations of $\mathcal{G}_A$, representation A defines the mapping from the initial mesh $\mathcal{M}_I$ to a modified mesh $\mathcal{M}'$:

$$\mathcal{R}_A(\mathcal{M}_I, \mathcal{P}_A, \mathcal{G}_A) : (\mathcal{M}_I, \mathcal{P}_A) \rightarrow (\mathcal{M}', \mathcal{P}'_A), \tag{11}$$

with $\mathcal{G}_A \in \mathbb{R}^{k_A}$ and $\mathcal{P}_A, \mathcal{P}'_A \in \mathbb{R}^{m_A \times 3}$.

*Representation B.* The second control volume is a standard representation resulting in a grid with $m_B = 64$ control points $\mathcal{P}_B$. In contrast to $\mathcal{R}_A$, the control volume is restricted to deformations of the upper chassis part only. Further constraints, which ensure the practicability of the designs are not included. For the parameterization of the control point grid, control points are effectively grouped into $k_B = 12$ groups, $\mathcal{G}_B = (CPG_1, \ldots, CPG_{k_B})^T$, as depicted in Fig. 9(a)–(c). Again, the modifications of the individual groups are restricted to displacements along distinct axis. In summary, given the $k_B = 12$ variable design parameters, the modification of the initial mesh utilizing representation B is defined by:

$$\mathcal{R}_B(\mathcal{M}_I, \mathcal{P}_B, \mathcal{G}_B) : (\mathcal{M}_I, \mathcal{P}_B) \rightarrow (\mathcal{M}', \mathcal{P}'_B), \tag{12}$$

with $\mathcal{G}_B \in \mathbb{R}^{k_B}$ and $\mathcal{P}_B, \mathcal{P}'_B \in \mathbb{R}^{m_B \times 3}$. Compared to representation A, the reduced mesh density allows larger variations of the car shape.

### 6.2. Computational fluid dynamics simulation

For each design that has been generated using FFD the aerodynamic performance is evaluated with a computational fluid dynamics solver. OpenFOAM[5] an open source CFD software package is used for simulating the flow around the passenger car surface. Therefore, the domain occupied by the flow is divided into discrete cells generating an octree based hexahedral CFD mesh with 3.3 million cells. Boundary conditions are specified, which define the flow behavior at the boundaries of the computational area, e.g. at the inlet or the design surface. A uniform flow with a velocity of 110 km/h is defined at the inlet of the flow domain. The Reynolds-averaged

[5] http://www.openfoam.com.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

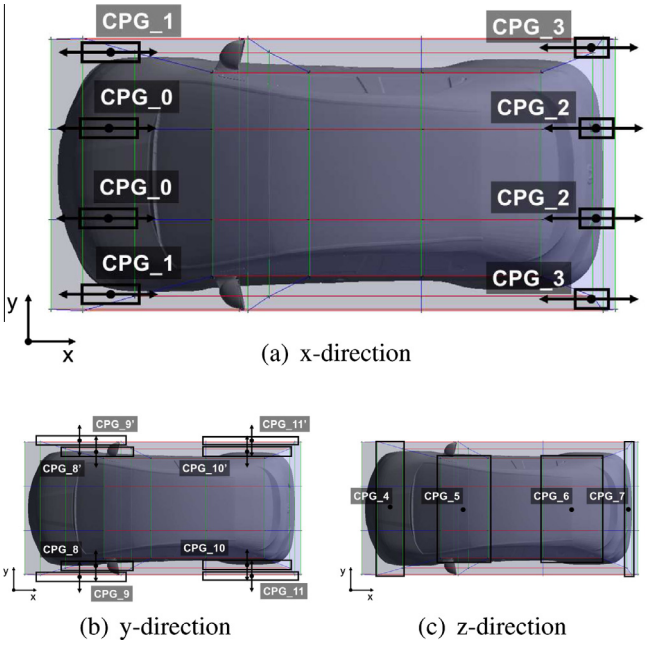



**Fig. 9.** Illustration of the 12 control point groups of **Representation B**.

Navier–Stokes equations are solved including the SST $k-\omega$ turbulence model, see [63].

Mainly two quantities are derived from the solution of the flow simulation, the overall drag force $F_D$ acting on the car in the direction of the freestream flow, and the rear lift $F_{LR}$, which is perpendicular to the fluid flow. $F_D$ and $F_{LR}$ are used to quantify the aerodynamic characteristics of each passenger car. Both measures define the objective of the design process, where a decrease in $F_D$ is directly linked to a reduction in the fuel consumption and a decrease in $F_{LR}$ to an improvement of the car stability.

### 6.3. Explorative search with latin hyper cube sampling

Sampling methods are widely used in engineering design in order to explore a constrained design space. Optimal sampling plan strategies are highly relevant for applications where full factorial experiments are infeasible due to high experimental costs. In engineering design, sampling plan methods are typically applied in order to produce properly distributed data for constructing an approximation model of the quality function. Subsequent optimization and design processes can utilize these approximation models as surrogates for expensive quality function evaluations. Among the most prominent and most frequently used sampling techniques are *Latin hypercube sampling (LHS)*, *Sobol sequences* and *orthogonal arrays*. In this study, an optimized LHS method, as described in [64] is used, which applies the optimality criteria of Morris and Mitchel [65] to achieve a space-filling sampling.

Given the representation $\mathcal{R}_A$, as described in Section 6.1, an optimal sampling of the $k_A = 16$ dimensional design space is targeted. The optimized LHS algorithm has been applied to generate a limited number of 500 design samples within a constrained design space. Each dimension is bounded between $-0.3$ and 0.3, which corresponds to a maximum displacement of each control point group by 0.3 m. The resulting samples are positioned in the center of each hyper cubic element. Utilizing $\mathcal{R}_A$, 500> modified instances of the baseline surface are generated. For all modified designs the airflow around the surface of the passenger car is simulated and its aerodynamic characteristics are calculated according to Section 6.2. Fig. 10 shows the resulting data. The relation between the drag force $F_D$ and the rear lift $F_{LR}$ is depicted in Fig. 10(a) and the shapes of two different non-dominated solutions are shown in Fig. 10(b) and (c).

### 6.4. Exploitive search with an evolutionary strategy

While sampling techniques target a uniform sampling of the entire design space, optimization algorithms like evolutionary strategies perform selective sampling along certain paths towards optimal solutions. Optimization algorithms often adapt their strategy parameters by exploiting information from previously generated solutions. In the following experiments, two optimization runs are carried out targeting the minimization of a pre-defined fitness function. A $(\mu, \lambda)$ evolutionary strategy with covariance matrix adaptation (CMA-ES) has been used for the optimization, see [66]. The process starts with the initialization of a population of $\mu$ parameter vectors. In each generation $g$, $\lambda$ solutions are sampled from a multi-variate normal distribution, $\vec{x}_i^{g+1} \sim \mathcal{N}\left(\langle\vec{x}\rangle^g, (\sigma^g)^2\mathbf{C}^g\right)$, $i = 1 \ldots \lambda$, around the mean $\langle\vec{x}\rangle^g$ of the $\mu$ so called parent solutions. After the fitnesses for all $\lambda$ solutions have been calculated, $\mu$ best out of $\lambda$ solutions are recombined to provide a new mean $\langle\vec{x}\rangle^{g+1}$ for the sampling in the subsequent generation. Besides the mean, the global step-size $\sigma^g$ and the covariance matrix $\mathbf{C}^g$ are updated according to Eqs. 2 to 5 of [67].

For the optimization of the passenger car a $(2, 12)$ CMA-ES has been applied using the implementation in the Shark machine learning library.[6] The $k_A = 16$ and $k_B = 12$ parameters of representation $\mathcal{P}_A$ and $\mathcal{P}_B$ span the search space for the two optimization runs, respectively. The $\mu = 2$ solutions are initialized with the baseline passenger car shape. The initial step size $\sigma^0$ is set to $\sigma^0 = 0.1$. The covariance matrix $\mathbf{C}^{(0)}$ is set to the unity matrix in the first generation. Thus, the first $\lambda = 12$ so called offspring are sampled from a uniform multi-variate normal distribution. Both optimization runs target the minimization of the overall drag force $F_D$ constrained by the rear lift $F_{LR}$, the volume $V$ and the maximum control point group displacements. This results in the following fitness function[7]:

$$f(\vec{x}) = F_D + \tau_1 \cdot p_1(\vec{x}) + \tau_2 \cdot p_2(V) + \tau_3 \cdot p_3(F_{LR})$$

$$p_1(\vec{x}) = \sum_{x_i} a,\ a = \begin{cases} 0 & \text{if } |x_i| \leqslant 0.3 \\ 1 & \text{if } |x_i| > 0.3 \end{cases}$$

$$p_2(V) = (V - V^c)^2$$

$$p_3(F_{LR}) = \begin{cases} 0 & \text{if } F_{LR} \leqslant F_{LR}^c \\ (F_{LR} - F_{LR}^c)^2 & \text{if } F_{LR} > F_{LR}^c \end{cases},$$

where $p_i$ and $\tau_i$ define the individual penalty terms and respective weightings. The values for $\tau_i$ are determined based on experience with $\tau_1 = 100, \tau_2 = 1000$ and $\tau_3 = 1$. With $V^c = 9.40\ \text{m}^3$, the generated meshes are expected to enclose a similar volume as the initial car. The upper bound for the rear lift $F_{LR}$ is set to $F_{LR}^c = 300.00$ N, allowing the rear lift to increase by about 16% compared to the baseline value of $F_{LR} = 252.53$ N. Furthermore, the search process should keep the control point displacements in a constrained range, punishing extreme deformations.

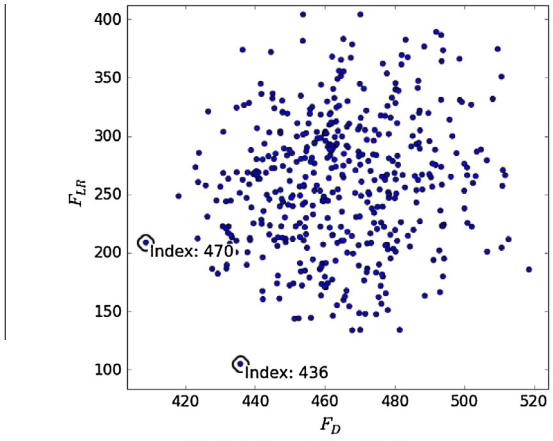
Two optimization runs have been carried out for 14 generations[8] based on $\mathcal{R}_A$ and $\mathcal{R}_B$, respectively. Each optimization run results in 168 different designs. The results of the two runs are summarized in Fig. 11. Fig. 11(a)–(d) visualize the progress of the fitness

[6] http://shark-project.sourceforge.net, [68].

[7] In the formulation of the quality function and its algorithmic realization, we ignore physical units and implicitly assume that the units of free parameters are chosen accordingly.

[8] In practice the number of generations is most often limited due to the high computational costs of the fitness evaluation. Improved designs can already be found using a lower number of generations, even the optimizer does not converge.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002




(a) Solutions in the objective space



(b) Car shape 470



(c) Car shape 436

**Fig. 10.** Results from the explorative search with LHS.

value, drag, volume and rear lift of the best design solution over the different generations, wherein the blue dashed line highlights the performance of the baseline car. Both optimization runs succeeded in developing car shapes that outperform the baseline. In most generations, the best solutions of both runs do not violate any of the volume and rear lift constraints. Especially in early generations, the optimization run based on $\mathcal{R}_B$ outperforms the optimization run using $\mathcal{R}_A$ with respect to the fitness and the achieved drag reduction. However, the designs from the run with $\mathcal{R}_A$ manage to achieve a better performance with respect to the rear lift. The advantage of $\mathcal{R}_B$ over $\mathcal{R}_A$ in the optimization results from the more severe constraints used for $\mathcal{R}_A$. This is apparent when comparing the shapes of the respective best designs, as depicted in Fig. 11(e) and (f). As can be seen, the optimization based on $\mathcal{R}_B$ results in designs with large deformations at the trunk of the car and mesh distortions at the back, ending up in an infeasible car shape.

For comparison to the results from the explorative search strategy, the position of all generated solutions in the overall objective space is shown in Fig. 11(g).

## 7. Passenger car meta design data

Each design that has been generated in one of the sub-processes 1–3 from the previous section, see Fig. 7, has been transferred into a surface mesh representation, with about $n = 550,000$ vertices. The respective performance values for $F_D$ and $F_{LR}$ have been linked to each design and its related mesh representation. From the entire set of designs, 4 meta design data sets are compiled, containing the designs from:

- **Data set 1**: optimized LHS with $\mathcal{R}_A$ (process 1).
- **Data set 2**: CMA-ES with $\mathcal{R}_A$ (process 2).
- **Data set 3**: CMA-ES with $\mathcal{R}_B$ (process 3).
- **Data set 4**: CMA-ES with $\mathcal{R}_A$ & $\mathcal{R}_B$ (process 2 & 3).

In each data set for each design $m$ the surface differences by means of the vertex displacements $\delta_{i,j}^{r,m}$ have been calculated with respect to the initial car shape, which has been chosen as reference design $r$. For simplicity, the Euclidean distance has been used to identify pairs of corresponding vertices. In addition to the displacement values, the differences in the design qualities have been quantified as well with respect to the reference design with $\phi_{F_D} = \phi_{F_D}^{r,m} = F_D^m - F_D^r$ and $\phi_{F_{LR}} = \phi_{F_{LR}}^{r,m} = F_{LR}^m - F_{LR}^r$.

In the first experimental studies, the design variations (relative to the baseline design $r$) within the different data sets have been visualized.

### 7.1. Identification of weakly deformed design areas

The analyses of the variances of local surface variations are shown in Fig. 12. For visualization, the calculated displacement variances are mapped as color values to the corresponding vertices of the reference design. Bluish areas indicate non or hardly deformed surface regions whereas reddish areas highlight those regions with a high variance.

From Fig. 12(a) and (b) it can be seen that the constraints on representation $\mathcal{R}_A$, e.g., at the front screen, result in regions of low variance. Comparing Fig. 12(a) and (b) one can note: while the LHS strategy targets an equal variation of all design parameters, the variations from the CMA-ES are restricted by the search path that the optimization algorithm follows. Fig. 12(d) depicts the results from the analysis of the combined data set.

A low variance might be assigned to a vertex for several reasons. Either a variation of a vertex was not possible due to limitations in the representation, e.g. due to hard constraints on the shape variations, or the variations were not realized in the search process. If the low variance is due to the representation of the design one might think about a change of the representation for any subsequent process. If the low variance is due to the course of the design process one might re-think the design strategy instead.

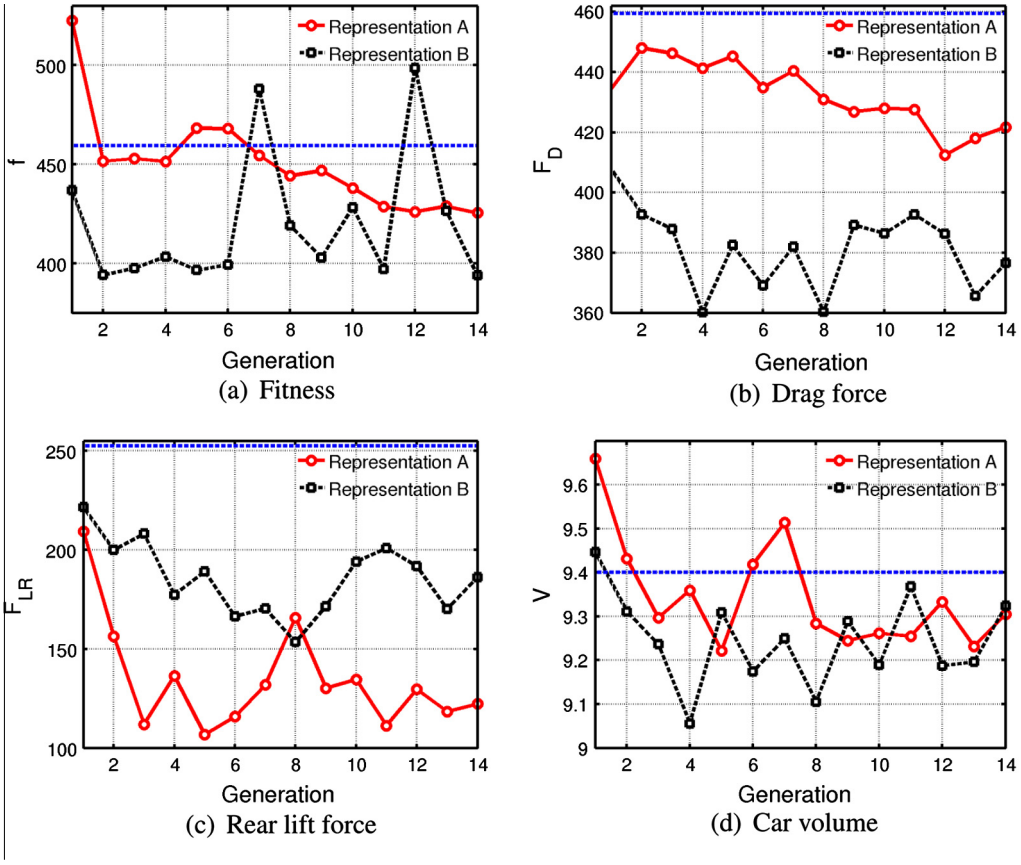
In our example, the variance analysis was carried out offline after the design processes were finished. However, it is equally possible to use the variance analysis as a monitoring tool during the search process to identify design regions which have been left unexplored.

### 7.2. Evaluation the course of design

Given the initial objectives and constraints of the design processes, each process follows a certain strategy to reach the design goals. However, whereas a clear strategy might be obvious for individual processes, for multiple sequential and parallel processes, where many engineers are involved, the actual direction of the overall design process might not be apparent to everyone. The analysis of the mean surface feature differences can depict information on the individual and combined strategies at the same time. As an example, estimating the arithmetic mean of the displacements for each vertex provides information on the global trend of the direction and the amount of surface modifications relative to the pre-defined reference design.

The resulting mean vertex displacements for the four example data sets are visualized in Fig. 13(a)–(d). Reddish (bluish) regions show that the mean displacement relative to the baseline surface is in (against) the direction of the surface normals, towards the outside (inside) of the car surface. Greenish regions are those where the average displacement is zero. Either, those regions have not been modified at all or the displacements in either direction have canceled each other out. Since the LHS targets an equal variation of the baseline design, the mean displacement value for each vertex vanishes, and an explicit direction or strategy is not visible, see Fig. 13(a). Small deviations from a zero mean displacement can be observed for individual vertices due to the non-linear transformation from the control point variations to variations of the surface points. Fig. 13(b) and (c) emphasize the overall direction of

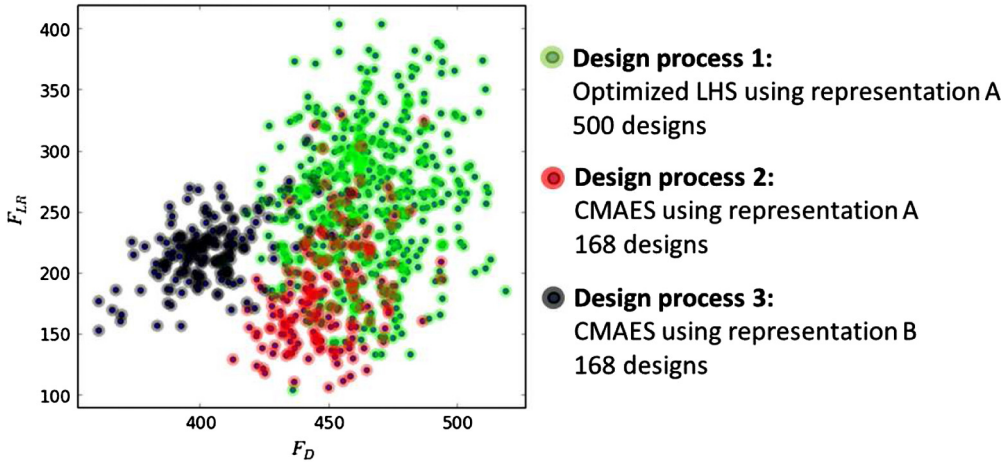
Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002






(e) Best shape using Representation A

(f) Best shape using Representation B




(g) Comparison of the solutions from all three design processes

**Fig. 11.** Illustration of the results from the optimization with the CMA-ES, using different design representations.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

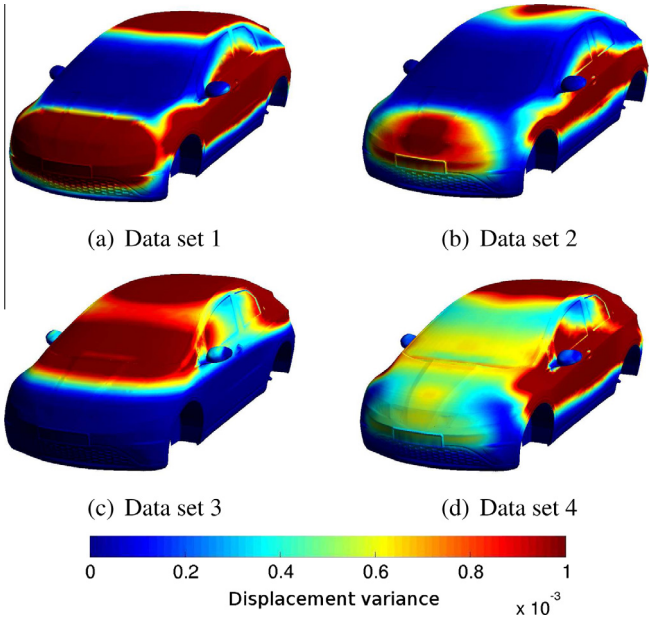



**Fig. 12.** Visualization of the displacement variance related to each vertex.

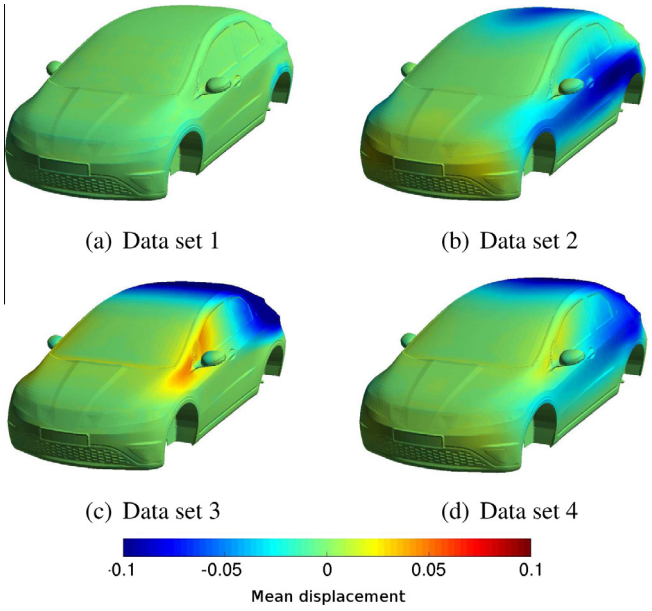



**Fig. 13.** Visualization of the mean displacement value related to each vertex.

the CMA-ES optimization runs. It can be observed in Fig. 13(b), that using $\mathcal{R}_A$, the surface areas colored in blue have been modified to the inside of the car surface. These modifications let to an improvement of the overall aerodynamic drag while complying with the constraints on the rear lift and the volume. For the optimization based on representation B ($\mathcal{R}_B$), the optimizer took a partly different strategy, see Fig. 13(c). While the rear of the car surface has also been modified towards the inside of the car, areas around the side mirror have been deformed to the outside.

The results show, that the analysis of the mean feature variations allow the observation of the trends of individual and combined, see Fig. 13(d), design processes. The interpretation and communication of the results can guide individual and global design strategies.

## 8. Passenger car sensitivity analysis

In the following experiments, sensitivity analysis has been applied to the design data in order to explicitly evaluate the relation between of local shape modifications $\delta_{i,j}^{r,m}$ and the objective values $\phi_{F_D}$ and $\phi_{F_{LR}}$.

### 8.1. Direction of performance improvement

Given the displacement data for each vertex of the reference mesh and the differences in the performance numbers, the sensitivities for a chosen reference design are calculated using the Pearson correlation coefficient. In addition to the correlation value, the statistical significance for each correlation value has been calculated. For visualization, both values have been mapped onto one color scale, where the actual color value is defined by the correlation value and the saturation of the color is defined by its significance. The results of the sensitivity analysis are depicted in Fig. 14(a)–(d), where Fig. 14(a)–(c) visualize the sensitivity to $\phi_{F_D}$ and Fig. 14(d) to $\phi_{F_{LR}}$ for different data sets. The interpretation of the correlation values has to be related to the surface features: Red (blue) areas indicate, that a modification of the surface into (against) the normal direction of the vertices will lead to an increase (decrease) of the performance indicator $F_D$ or $F_{LR}$. Areas of low correlation, i.e. without significant effect on the performance are shown in green. For regions with low saturation (white areas) no conclusion can be drawn since the statistical significance of the correlation is low.

From the analysis of the sensitivity results in Fig. 14(a)–(c) one can derive the basic rule that a deformation of the rear part to the inside of the car will reduce the drag and thus improve the car performance. While this concept is likely to be known to the aerodynamic engineer, the information that the deformation of the area close to the front door of the car towards the outside of the initial car surface can lead to an reduction of the drag might denote a more interesting relation. Furthermore, the analysis of the joint data set (see Fig. 14(c)) results in drag sensitivities at the outer front bumper, which are less obvious from the analysis of the individual data sets. Figs. 14(c) and (d) allow the comparison between drag and rear lift sensitivities. A clear trade-off between $F_D$ and $F_{LR}$ can be observed for the region around the passenger's door. A deformation of those surface patches to the outside is expected to result in a lower drag value. However, such a modification would also result in an increase in the rear lift.

In general, it should be noted that there is always the chance that high correlations can result from unresolved co-variances or from outliers in the data, and thus a verification of the most interesting sensitivities should be obligatory.

### 8.2. Reliability of probabilistic sensitivity estimates

Mutual information or its robust variant, see [25] for reference, provide an alternative and more general approach to the sensitivities estimation, of which compared to the correlation coefficient make no assumption on the kind of interrelation between design modifications and performance changes. However, for a reliable estimation a sufficient number of data samples needs to be available for analysis. Based on the given passenger car design data the dependency of the mutual information and the robust mutual information on the size of the design data set is studied. Given the data set 4, the sensitivities are calculated for different sample sizes $K \in \{100, 200, 300\}$. Instead of randomly selecting $K$ designs from the data set, the $K$ designs closest to the baseline shape are identified based on the calculation of the average absolute displacement.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

The resulting sensitivities using the mutual information and the robust mutual information are summarized in Fig. 15. Red areas show surface patches with high sensitivity and blue areas are those that have no influence on the drag value. The results of the two measures for $K = 300$ are qualitatively similar. If $K$ is reduced, the mutual information fails in clearly separating sensitive from insensitive areas, in particular for $K = 100$. The results show that especially for the analysis of small design data sets the robust mutual information should be preferred to the classical mutual information.

### 8.3. Sensitive areas

The number of vertices that defines the shape of each design in the data sets is around $n = 550,000$. A reliable modeling of the interrelations between distant design regions based on all vertices is hardly possible. Therefore, the feature reduction method from Section 3.2 has been applied to pre-select the most relevant features from the data set. In order to form sensitive areas along the baseline passenger car surface the sensitivity information from the Pearson correlation analysis, see Fig. 14(d), has been used. Together with the geodesic distance, the sensitivity values define the similarity between vertices. In a pre-processing step vertices with a low sensitivity, $|r^r_{\Delta_i}| < 0.6$, have been removed before the clustering step. Thereafter, the K-Means clustering algorithm is used for the formation of sub-areas on the surface. The results of the clustering are illustrated in Fig. 16(a) and (b), where vertices with the same color value define one cluster. Vertices with low sensitivity have been neglected and are shown in dark blue. With $k = 12$, the number of clusters has been pre-defined. Vertices closest to their centers define the cluster centers.

As can be seen from the results, the sensitive areas are not symmetric along the center line, while the surface mesh is. Such symmetry requirements need to be explicitly incorporated into the clustering algorithm. This has not been considered so far and remains future work. For simplicity, a subset of $k = 6$ cluster centers has been selected manually from the entire set, neglecting centers which represent identical design areas. The selected centers are labeled in Fig. 16, whereas the sensitive areas are labeled from $A0$ to $A5$. In addition the vertex index of the cluster center is given. Thus, e.g. $v129125$ defines the cluster center of area $A2$. The displacement values of the extracted cluster centers are the basis for the identification of car concepts and the investigation of higher order interaction patterns between distant design areas.

## 9. Passenger car design concept retrieval

In the following experimental studies, design concepts are extracted by applying two machine learning techniques, the tree learner and the self-organizing map to the reduced data set. Target is the formulation of abstract classes of passenger cars, which are similar with respect to their surface representation and their aerodynamic quality. For each concept crisp rules are extracted in human readable form.

### 9.1. Tree induction for car concept retrieval

The applied tree learner is an implementation of the standard C4.5 tree induction algorithm [40]. The gain ratio has been applied to split the data samples at each node and thus grow the tree. If the gain ratio is lower than 0.2 or the number of samples represented by one node is below three the growth has been stopped. In addition, each node is restricted to binary splits. The independent parameters for the induction algorithm capture the displacement values of the cluster centers that represent the $k = 6$ sensitive areas $A0, \ldots, A5$. The gain ratio for the split of the nodes is calculated based on a single scalar value. Therefore, the aerodynamic drag and the rear lift force have been combined into one characteristic value $\phi_F$, which is defined as the product between the normalized force differences:

$$\phi_F = \hat{\phi}_{F_{LR}} \cdot \hat{\phi}_{F_D}, \tag{13}$$

where $\hat{\phi}_{F_{LR}}$ and $\hat{\phi}_{F_D}$ define the respective normalized drag and rear lift force difference. The difference value $\phi_F$ has been calculated for each design of data set 4 prior to the tree induction.

The resulting tree is visualized in Fig. 17(b), where each node represents one potential car concept. The labels on top of each

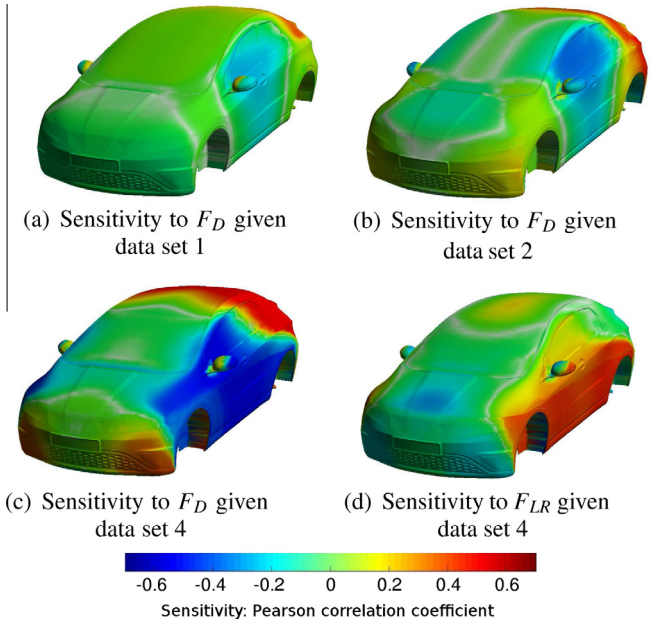



**Fig. 14.** Illustration of the results from the sensitivity analysis based on the calculation of the Pearson correlation.

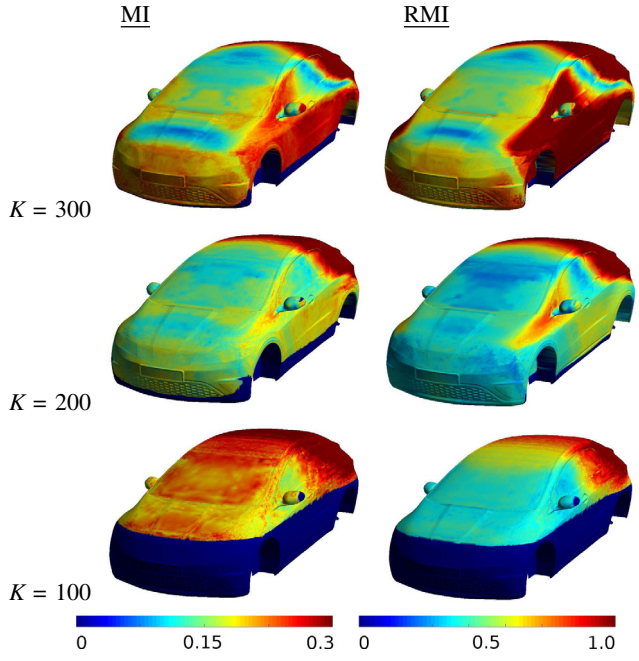



**Fig. 15.** Comparison of the sensitivity analysis using mutual information (left) and the robust variant of the mutual information (right).

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

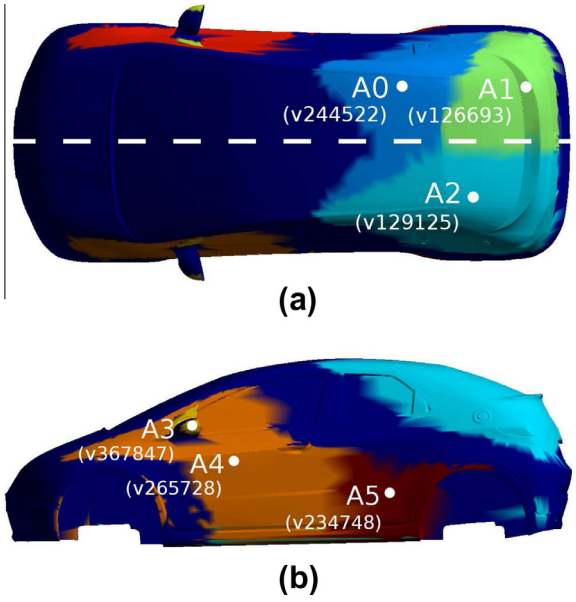



**Fig. 16.** Results of the feature reduction step by means of identifying sensitive areas and related cluster centers.

node capture the average value of $\phi_F$ and the label of the split variable (not present in the terminal nodes). The split variable $v244522$ in the root node, representing deformations of the roof among area ($A0$), defines the variable that best splits the entire set of designs with respect to $\phi_F$. Three nodes $A, B$ and $C$ have been selected to describe the underlying concepts in linguistic form, such as:

$$\begin{aligned}
A &: A0 \in [-0.29, -0.20] \wedge A2 \in [-0.27, -0.24] \rightarrow \\
&\quad \phi_{F_D} \in [-111.25, -102.40] \wedge \phi_{F_{LR}} \in [-99.50, -66.31], \\
B &: A0 \in [-0.29, -0.20] \rightarrow \\
&\quad \phi_{F_D} \in [-111.25, -59.65] \wedge \phi_{F_{LR}} \in [-99.50, -44.10], \\
C &: A0 \in [-0.07, -0.04] \wedge A5 \in [-0.11, -0.06] \rightarrow \\
&\quad \phi_{F_D} \in [-49.83, -5.81] \wedge \phi_{F_{LR}} \in [-140.92, -31.91].
\end{aligned}$$

The designs covered by each of the three rules are depicted with respect to their quality values in Fig. 17(a). The rules describe general coherences between surface deformations with respect to the reference design and the expected objective values. As an example, rule $B$ describes that a deformation of the surface patch around the roof (area $A0$) to the inside of the surface between 29 and 30 cm will lead to a respective drag and rear lift reduction of $-111.25$ N to $-59.65$ N and $-99.50$ N to $-44.10$ N. Especially for more complex or multiple trees generated from different data sets an analysis of all nodes can get laborious and the ranking or filtering of concepts is needed. For this purpose the utility measure, as introduced in Section 4.2, is used to rank concepts according to their relevance. In order to emphasize the benefit of the utility measure, concepts $A$ to $C$ have been evaluated accordingly. The results are summarized in Table 2.

The concept evaluation assigns the highest utility value to concept $A$ followed by $B$ and $C$. The hyper-volumes of concepts $A$ and $B$ are the same, since both concepts share the non-dominated design solutions. However, the $IS$ value is lower for concept $B$ and thus, concept $B$ is of lower utility compared to $A$.

### 9.2. Car concept learning with SOMs

As an alternative to the tree learning, an unsupervised partitioning of the design data has been carried out using the SOM algorithm in order to derive new design concepts. The input parameters for the artificial neural network are again the parameters capturing the displacements of areas $A0$–$A5$. In order to perform a supervised clustering with the SOM algorithm, the performance number $\phi_F$ (see Eq. 13) has been added as an additional input. The weight vectors of the $5 \times 5$ feature map are adapted based on the Euclidean distance between the input samples. After learning, each weight vector represents one prototype vector for one potential design concept. The topographic map, colored with the weight values of $A0$, is shown in Fig. 18(d). In addition, the U-Matrix has been calculated and is depicted in Fig. 18(e). The U-Matrix shows the distance between the weight vectors in the input space and can provide information to further group individual concepts into more general ones. Three concepts $D, E$ and $F$ have been selected and the corresponding designs have been denoted in the objective space in Fig. 18(f). The linguistic rules are as follows:

$$\begin{aligned}
D &: A0 \in [-0.29, -0.18] \wedge A1 \in [-0.31, -0.22] \wedge \\
&\quad A2 \in [-0.27, -0.20] \wedge A3 \in [0.00, 0.00] \wedge \\
&\quad A4 \in [0.00, 0.03] \wedge A5 \in [0.00, 0.00] \rightarrow \\
&\quad \phi_{F_D} \in [-111.25, -79.17] \wedge \phi_{F_{LR}} \in [-99.50, -66.31], \\
E &: A0 \in [-0.11, -0.06] \wedge A1 \in [-0.05, 0.03] \wedge \\
&\quad A2 \in [-0.05, 0.01] \wedge A3 \in [-0.02, 0.00] \wedge \\
&\quad A4 \in [-0.07, 0.00] \wedge A5 \in [-0.09, -0.05] \rightarrow \\
&\quad \phi_{F_D} \in [-59.02, -35.05] \wedge \phi_{F_{LR}} \in [-141.63, -82.59], \\
F &: A0 \in [-0.22, -0.15] \wedge A1 \in [-0.26, -0.17] \wedge \\
&\quad A2 \in [-0.24, -0.16] \wedge A3 \in [0.00, 0.00] \wedge \\
&\quad A4 \in [-0.01, 0.03] \wedge A5 \in [0.00, 0.00] \rightarrow \\
&\quad \phi_{F_D} \in [-98.38, -64.97] \wedge \phi_{F_{LR}} \in [-69.58, -26.89],
\end{aligned}$$

Here, the parameter range of each design parameter is used to formulate the antecedent part of the rule. Typically, parameters with vanishing variances can be removed, requiring an additional post-processing step.

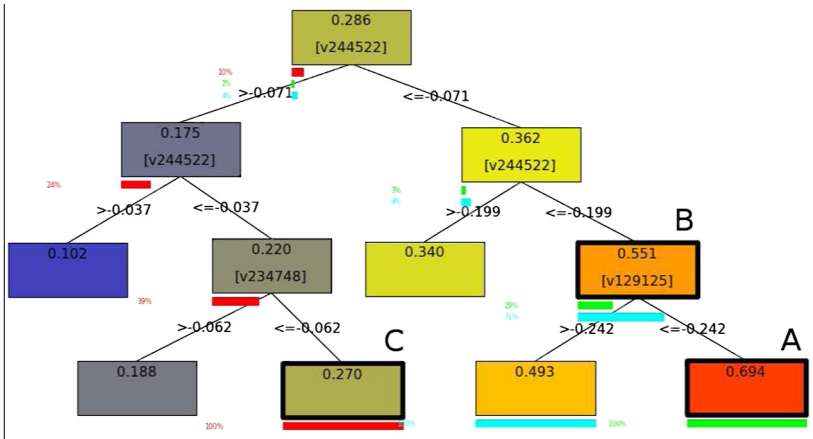
All in all, the SOM represents 25 concepts. In order to rank the individual concepts the utilities have been calculated for each concept and visualized in a *utility map*, see Fig. 18(a). Without analyzing the topographic maps of all objective values, the utility map provides a quick visual summary of the most relevant concepts. The three concepts $D, E$ and $F$ are the ones with the highest utilities. For a detailed analysis, the $IS$ and expected hyper volume for each concept are visualized in Fig. 18(b) and (c). As can be seen, concept $E$ has a higher correlation between design parameter values and the objective values, but a lower hyper volume compared to concept $D$.

Using the tree induction, SOMs or alternative techniques from machine learning and data mining engineers can extract new concepts explaining relations in the design domain. By adding the concepts to a global knowledge base the knowledge can be shared among engineers and utilized to improve future designs. Wherein, the utility measure can help to sort concepts according to each engineer's objectives, as shown in our experimental studies.

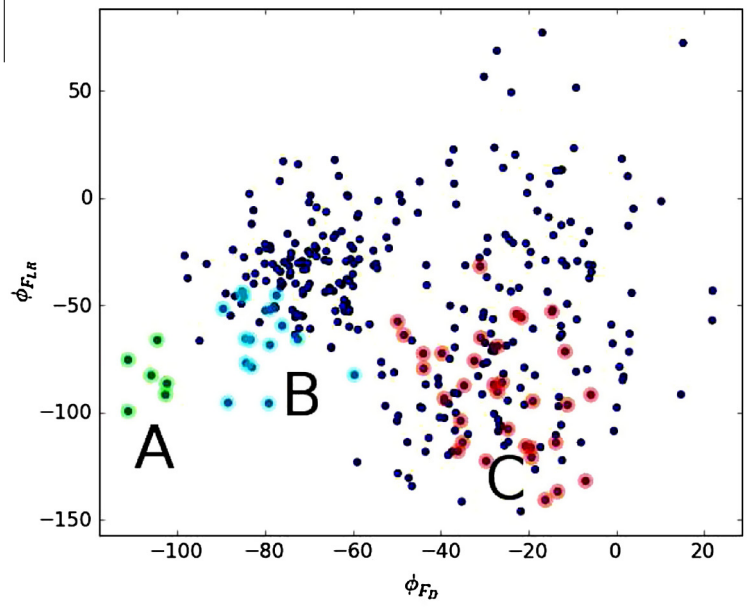
## 10. Passenger car interaction analysis

The interaction analysis, described in Section 5, is applied to the data set from design process 2 and 3. Each design is defined by the displacement values of the vertices representing the sensitive areas, $A0$–$A5$. Based on these data the two and three way interactions between the parameter and objective values have been calculated. Three objective values $\phi_{F_D}, \phi_{F_{LR}}$ and $\phi_F$ (Eq. (13)) are considered for the interaction analysis. The calculated interaction values are normalized by the entropy of the objectives with

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002




(a) Decision tree as hierarchical concept representation




(b) Illustration of the solutions covered by the selected concepts

**Fig. 17.** Result from the concept formation using tree induction. Three concepts $A, B$ and $C$ are selected for a detailed description.

$H(\phi_{F_D}) = 2.98, H(\phi_{F_{LR}}) = 2.86$ and $H(\phi_F) = 2.73$. For estimating the marginal distributions the parameter values are discretized into $b = 10$ bins. The estimation of the joint distributions has been carried out accordingly. The results of the interaction analysis in form of interaction graphs are depicted in Fig. 19(a)–(c). For each node the label ($A0$–$A5$) and the mutual information between the parameter and the quality values are shown. The edges between two nodes depict qualitatively the interaction between two parameters and the objective values. Solid lines denote synergy and dashed lines redundant interrelations between the parameters. The thickness of each line corresponds to the strength of the interaction. Edges with an interaction value below 0.05 are classified as irrelevant and are not visualized in the graph.

Fig. 19(a) depicts the interaction graph for the drag force. With a relative mutual information value of 0.465 parameter $A2$, representing variations at the rear window, reduces already a large portion of the uncertainty about the drag. Thus, knowing only $A2$ would already allow to correctly predict the discrete drag value for nearly 50% of all designs. All evaluated three-way interactions between parameters result in negative interaction information denoting a high degree of redundancy between the parameters. For example, analyzing the interaction between $A2$ and $A1$, this indicates that variations in $\phi_{F_D}$ due to $A2$ can equally result from variations of $A1$. Thus, $A1$ can implement an alternative control parameter in the design process, e.g. if variations of $A2$ are constrained.

**Table 2**
Detailed results from the concept evaluation for concepts $A, B$ and $C$, derived from the decision tree.

| | #designs | IS | vol | util |
|---|---|---|---|---|
| A: | 6 | 1.00 | 0.79 | **0.79** |
| B: | 21 | 0.65 | 0.79 | **0.51** |
| C: | 33 | 0.52 | 0.48 | **0.25** |

The interaction graph for the rear lift force is shown in Fig. 19(b). Parameter $A5$, which represents modifications at the back side of the car, has the highest sensitivity. For this parameter no significant interactions with other parameters have been observed. Thus, for the optimization of the rear lift, area $A5$ can be modified independently. In contrast, the results of the analysis of the three-way interaction indicate that variations of $A0$–$A4$ have to be jointly considered in the process of optimization.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

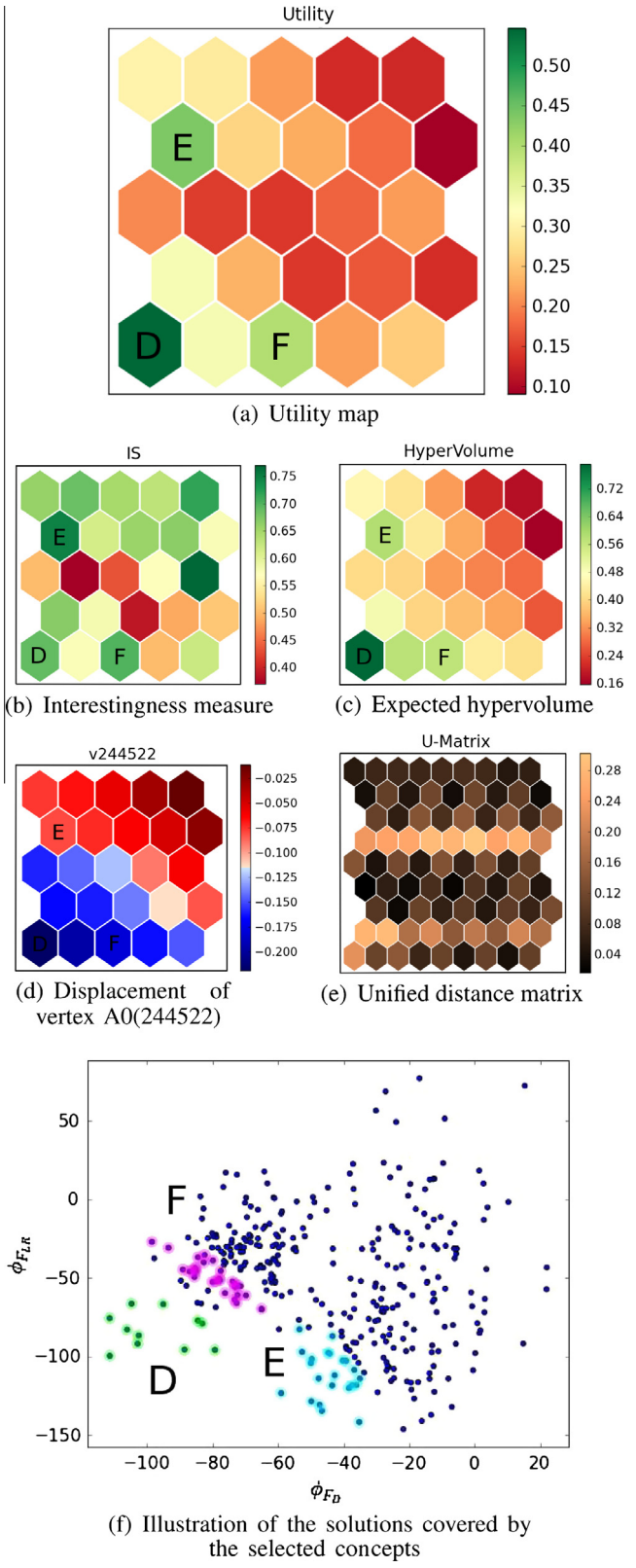



**Fig. 18.** Result from the concept formation using SOM. Three concepts *D*, *E* and *F* are selected for a detailed description.

In order to analyze the interrelations between design parameters and the combination of drag and rear-lift force, the interaction graph with respect to $\phi_F$ is shown in Fig. 19(c). The most relevant parameters are *A*0–*A*2 representing areas along the roof. Interestingly, a relevant interaction has been discovered between *A*1 and *A*4, which represent modifications at the back and the front side of the car, respectively. Considering such interactions during the design process might per se not be obvious.

These examples show that the interaction analysis can be a helpful tool to identify and characterize relevant interrelations between design parameters and objectives.

## 11. Summary and conclusion

In this paper, we have motivated the need for holistic data analytics in engineering design and outlined a framework for its

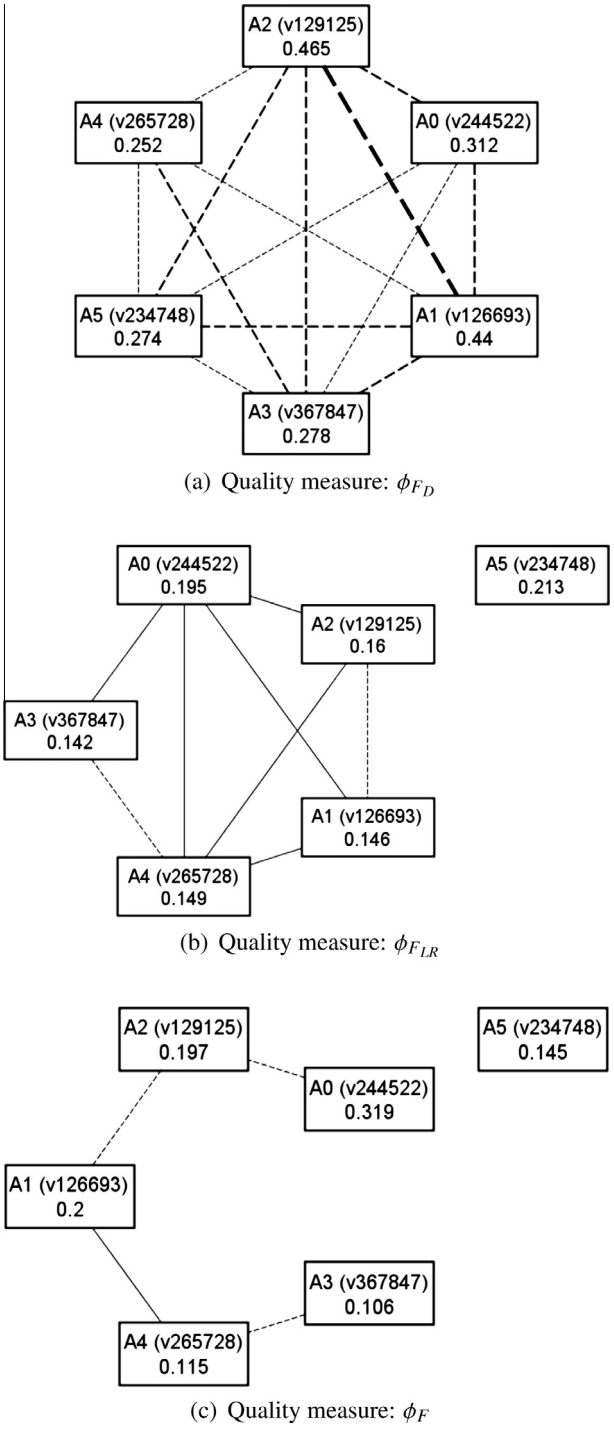



**Fig. 19.** Interaction graphs, illustrating the interrelation between distant surface areas with respect to different quality measures.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

realization. In order to be able to combine information from different design processes into one framework, a unified design representation has to be defined. The unified surface data is stored in a database, which is the central element of the framework as depicted in Fig. 4. The adaptation of statistical data mining methods to surface data is preceded by the definition of appropriate distance measures between geometrical objects. Even though the surface differences considered in our research are not large, we have shown that the Geodesic distance is more suitable than the standard Euclidean distance between surface vertices. The sensitivity analysis based on correlation methods and information theoretic approaches applied to surface data constitutes the first step towards *shape mining*.

The application of more sophisticated methods of knowledge formation necessitates to resolve the typical drawback of the universal design representation, i.e., its high dimensionality. Therefore, we apply feature evaluation, reduction and clustering techniques to reduce the representation to a significant subset. Based on this feature set, methods for concept retrieval can be applied. The procedure for the retrieval, description and evaluation of design concepts has been generalized and can be carried out independently of the used modeling technique. A new measure has been introduced to evaluate extracted design concepts based on the estimation of their utility. The new measure allows the ranking of concepts according to the formulation of the engineer's objectives.

The last analytics step in our framework as shown in Fig. 4 is the interaction analysis. The extension of mutual information to more than two random variables is non trivial and its comprehensive statistical treatment is beyond the scope of this paper. Nevertheless, we are able to formulate the statistical interaction between the changes of different surface patches or features and changes in the design objectives. The resulting interaction graphs provide a fast and easy way to visualize rather complex statistical information.

The remaining component in the proposed *shape mining* framework in Fig. 4 is the *utilization of the extracted knowledge*. Although it is impossible to directly show how the result of *shape mining* influences a realistic design process involving different tools, engineers and decision making processes, we apply the framework to the practical example of passenger car design. We choose two meaningful objectives and generate data from three different processes involving different shape representations and different strategies for shape space sampling. In the second half of the paper, we go through the different steps of our framework using these three realistic data sets.

The findings of the displacement and sensitivity analysis provide the engineer with a concise picture of the direction of the overall design process. The effects of different representations and sampling techniques can be visualized and unexplored regions of the design space can be identified. Whether those "white spots" on the design landscape are due to constraints or due to shortcomings of the design processes has to be decided by the engineer. Although the statistical methods to extract this information are not complex, it is the comprehensive framework that allows their application to the complete process. In a practical engineering design process for complex products like automobiles the impact of such rather basic information should not be underestimated.

The concept retrieval augmented by the new utility measure allows the formulation of simple rules for passenger car design matched to the specific objectives that have been formulated. The key here are the rules of intermediate complexity because they are still readable and are less likely to represent standard engineering knowledge for car design. Furthermore, the algorithmic extraction of design concepts allows the wider distribution of subjective engineering knowledge in a company.

Finally, the interaction analysis revealed the joint influence of modifications at the front and the back of the car on the accumulated objective $\phi_F$. This is likely to be new and interesting for the engineer. Whereas experienced engineers are quite confident to judge the interaction between design changes for single objectives, this is usually not the case if objectives are accumulated or when optimal trade-offs between objectives are sought. At the same time, design processes based on single objectives become more and more obsolete and are replaced by multi-disciplinary and many-objective approaches.

## Acknowledgments

The authors would like to thank Markus Olhofer for his valuable comments and advice, Stefan Menzel for the support in setting up the FFD representation and Giles Endicott for sharing his knowledge in fluid dynamics and for providing the LHS code (all from the Honda Research Institute Europe). Furthermore, we would like to thank Yusuke Uda from Honda R&D Co., Ltd for his contributions.

## References

[1] T. Fasoli, S. Terzi, E. Jantunen, J. Kortelainen, J. Sski, T. Salonen, Challenges in data management in product life cycle engineering, in: J. Hesselbach, C. Herrmann (Eds.), Glocalized Solutions for Sustainability in Manufacturing, Springer, Berlin Heidelberg, 2011, pp. 525–530.
[2] O. Eck, D. Schaefer, A semantic file system for integrated product data management, Advan. Eng. Inform. 25 (2011) 177–184.
[3] L.D. Xu, Enterprise systems: state-of-the-art and future trends, IEEE Trans. Indust. Inform. 7 (2011) 630–640.
[4] G. Pahl, W. Beitz, J. Feldhusen, K.H. Grote, Engineering Design, third ed., A Systematic Approach, Springer, 2007.
[5] K. Chiba, S. Jeong, S. Obayashi, H. Morino, Data mining for multidisciplinary design space of regional-jet wing, IEEE Cong. Evolut. Comput. 3 (2005) 2333–2340.
[6] S. Obayashi, S. Jeong, K. Shimoyama, K. Chiba, H. Morino, Multi-objective design exploration and its applications, Int. J. Aeronaut. Space Sci. 11 (2010) 247–265.
[7] M. Rath, L. Graening, Modeling design and flow feature interactions for automotive synthesis, in: International Conference on Intelligent Data Engineering and Automated Learning (IDEAL), Springer, 2011.
[8] L. Graening, S. Menzel, M. Hasenjäger, T. Bihrer, M. Olhofer, B. Sendhoff, Knowledge extraction from aerodynamic design data and its application to 3d turbine blade geometries, Math. Model. Algor. 7 (2008) 329–350.
[9] S. Wilkinson, S. Hanna, L. Hesselgren, V. Mueller, Inductive aerodynamics, in: Computation and Performance – Proceedings of the 31st eCAADe Conference, vol. 2, pp. 39–48.
[10] E.H. Spanier, Algebraic Topology, MMcGraw-Hill, New York, 1966.
[11] M. Alexa, Recent advances in mesh morphing, Comp. Graph. Forum 21 (2002) 173–197.
[12] A. Bronstein, M. Bronstein, R. Kimmel, Numerical Geometry of Non-Rigid Shapes, first ed., Springer Publishing Company, Incorporated, 2008.
[13] T. Fukushige, T. Ariyoshi, Y. Uda, Y. Okuma, T. Okabe, K. Yamamoto, L. Graening, S. Menzel, M. Olhofer, Development of real-time aerodynamic design system, Honda R&D Tech. Rev. 24 (2012) 107–116.
[14] Y. Wang, B.S. Peterson, L.H. Staib, Shape-based 3d surface correspondence using geodesic and local geometry, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 644–651.
[15] P.J. Besl, N.D. McKay, A method for registration of 3-d shapes, IEEE Trans. Pattern Anal. Mach. Intell. 14 (1992) 239–256.
[16] T. Jost, Fast Geometric Matching for Shape Registration, Ph.D. thesis, Université de Neuchȩtel, 2002.
[17] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, Global Sensitivity Analysis, The Primer, John Wiley & Sons, 2007.
[18] M.B. Giles, N.A. Pierce, An introduction to the adjoint approach to design, Flow, Turbul. Combust. 65 (2000) 393–415.
[19] C. Othmer, Cfd topology and shape optimization with adjoint methods, VDI-Berichte, in: 13th Internationaler Kongress, Berechnung und Simulation im Fahrzeugbau, vol. 1967, 2006, pp. 61–72.
[20] A. Verma, An introduction to automatic differentiation, Curr. Sci. 78 (2000) 804–807.
[21] J. Ascough, G. Thimoty, M. Liwang, A. Lajpat, Key criteria and selection of sensitivity analysis methods applied to natural resource models, in: International Congress on Modeling and Simulation Proceedings, Salt Lake City, UT.
[22] A. Saltelli, S. Tarantola, K.P.-S. Chan, A quantitative model-independent method for global sensitivity analysis of model output, Technometrics 41 (1999) 39–56.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002

[23] I. Sobol', Sensitivity estimates for nonlinear mathematical models, Math. Model. Comput. Exper. 1 (1993) 407–414.
[24] G.C. Critchfield, K.E. Willard, D.P. Connely, Probabilistic sensitivity analysis methods for general decision models, Comput. Biomed. Res. 19 (1986) 254–265.
[25] L. Graening, S. Menzel, T. Ramsay, B. Sendhoff, Application of sensitivity analysis for an improved representation in evolutionary design optimization, in: IEEE International Conference on Genetic and Evolutionary Computing (ICGEC), IEEE Press, 2012, pp. 1–4.
[26] E.W. Dijkstra, A note on two problems in connexion with graphs, Numer. Math. 1 (1959) 269–271.
[27] A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review, ACM Comput. Surv. 31 (1999) 264–323.
[28] D. Pelleg, A. Moore, X-means: Extending k-means with efficient estimation of the number of clusters, in: Proceedings of the Seventeenth International Conference on Machine Learning, Morgan Kaufmann, San Francisco, 2000, pp. 727–734.
[29] R. Tibshirani, G. Walther, T. Hastie, Estimating the number of clusters in a data set via the gap statistic, J. Roy. Statist. Soc. Ser. B 63 (2001) 411–423.
[30] W.M. Hsu, J.F. Hughes, H. Kaufman, Direct manipulation of free-form deformations, in: SIGGRAPH '92: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques, ACM Press, New York, NY, USA, 1992, pp. 177–184.
[31] M. Kantardzic, Data Mining – Concepts, Models, Methods, and Algortihms, IEEE Press, 2001.
[32] R.J. Miller, Y. Yang, Association rules over interval data, in: SIGMOD Conference'97, pp. 452–461.
[33] L. Geng, H.J. Hamilton, Interestingness measures for data mining: a survey, ACM Comput. Surv. 38 (2006).
[34] G.I. Webb, Discovering significant rules, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '06, ACM, New York, NY, USA, 2006, pp. 434–443.
[35] P. Tan, V. Kumar, Interestingness measures for association patterns: a perspective, in: KDD 2000 Workshop on Postprocessing in Machine Learning and Data Mining, Boston, MA.
[36] K. Deb, Multi-Objective Optimization Using Evolutionary Algorithms, first ed., Wiley, 2001.
[37] L. While, L. Bradstreet, L. Barone, A fast way of calculating exact hypervolumes, IEEE Trans. Evolut. Comput. 16 (2012) 86–95.
[38] M.T. Mitchell, Machine Learning, McGraw-Hill, New York, 1997.
[39] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, Pattern Classification and Scene Analysis: Pattern Classification, Wiley, 2001.
[40] L. Breiman, J. Friedman, R. Olshen, C. Stone, Classification and Regression Trees, Wadsworth and Brooks, Monterey, CA, 1984.
[41] Z. Pawlak, Rough Sets, Theoretical Aspects of Reasoning About Data, Kluwer, Dordrecht, 1991.
[42] L.A. Zadeh, G.J. Klir, B. Yuan, Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers, Advances in Fuzzy Systems, World Scientific, 1996.
[43] G.P. Zhang, Neural networks for classification: a survey, IEEE Trans. Syst., Man Cybernet., Part C (Appl. Rev.) 30 (2000) 451–462.
[44] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (1995) 273–297.
[45] S.R. Safavian, D. Landgrebe, A survey of decision tree classifier methodology, IEEE Trans. Syst., Man Cybernet. 21 (1991) 660–674.
[46] L. Rokach, M. Oded, The Data Mining and Knowledge Discovery Handbook, Springer, pp. 321–352.
[47] T. Kohonen, Self-Organizing Maps, Springer, Heidelberg, 1995.
[48] J.A. Kangas, T.K. Kohonen, J.T. Laaksonen, Variants of self-organizing maps, IEEE Trans. Neural Netw. 1 (1990) 93–99.
[49] T.K. Kohonen, E. Oja, O. Simula, A. Visa, J. Kangas, Engineering applications of the self-organizing map, Proc. IEEE 84 (2002) 1358–1384.
[50] S. Obayashi, D. Sasaki, Visualization and data mining of pareto solutions using self-organizing map, in: Proceedings of the 2nd International Conference on Evolutionary Multi-Criterion Optimization, EMO'03, Springer-Verlag, Berlin, Heidelberg, 2003, pp. 796–809.
[51] B. Hammer, A. Rechtien, M. Strickert, T. Villmann, Rule extraction from self-organizing networks, in: Lecture Notes in Computer Science, Springer, 2002, pp. 877–882.
[52] A. Ultsch, Knowledge extraction from self-organizing neural networks, Inform. Classif. (1993) 301–306.
[53] J. Malone, K. Mcgarry, S. Wermter, C. Bowerman, Data mining using rule extraction from Kohonen self-organising maps, Neural Comput. Appl. 15 (2006) 9–17.
[54] G.L. Rocca, Knowledge based engineering: between ai and cad. review of a language based technology to support engineering design, Advan. Eng. Inform. 26 (2012) 159–179.
[55] S. Kern, S.D. Müller, N. Hansen, D. Büche, J. Očenášek, P. Koumoutsakos, Learning probability distributions in continuous evolutionary algorithms – a comparative review, Nat. Comput. 3 (2004) 77–112.
[56] K. Krippendorff, Information Theory: Structural Models for Qualitative Data, Sage University Paper Series on Quantitative Applications in the Social Sciences, vol. 07-062, Sage, Newbury Park, CA, 1986.
[57] W.J. McGill, Multivariate information transmission, Psychometrika 19 (1954) 97–116.
[58] A. Jakulin, I. Bratko, Testing the significance of attribute interactions, in: R. Greiner, D. Schuurmans (Eds.), Proceedings of the Twenty-First International Conference on Machine Learning (ICML-2004), Banff, Canada, pp. 409–416.
[59] K. Krippendorff, Ross ashby's information theory: a bit of history, some solutions to problems, and what we face today, Int. J. Gen. Syst. 38 (2009) 189–212.
[60] A. Jakulin, Machine Learning Based on Attribute Interactions, Dissertation, 2005.
[61] T.W. Sederberg, S.R. Parry, Free-form deformation of solid geometric models, SIGGRAPH Comput. Graph. 20 (1986) 151–160.
[62] S. Menzel, B. Sendhoff, Representing the change – free form deformation for evolutionary design optimization, in: T. Yu, L. Davis, C.M. Baydar, R. Roy (Eds.), Evolutionary Computation in Practice, Studies in Computational Intelligence, vol. 88, Springer, 2008, pp. 63–86.
[63] F.R. Menter, Zonal two-equation $k$–$\omega$ turbulence models for aerodynamic flows, AIAA J. (1993) 2906.
[64] A. Forrester, A. Sobester, A. Keane, Engineering Design via Surrogate Modelling: A Practical Guide, Wiley, 2008.
[65] M.D. Morris, T.J. Mitchell, Exploratory designs for computational experiments, J. Statist. Plann. Infer. 43 (1995) 381–402.
[66] N. Hansen, A. Ostermeier, Completely derandomized self-adaptation in evolutionary strategies, Evolut. Comput. 9 (2001) 159–195.
[67] N. Hansen, S. Kern, Evaluating the cma evolution strategy on multimodal test functions, in: X. Yao et al. (Eds.), Parallel Problem Solving from Nature PPSN VIII, LNCS, vol. 3242, Springer, 2004, pp. 282–291.
[68] C. Igel, V. Heidrich-Meisner, T. Glasmachers, Shark, J. Mach. Learn. Res. 9 (2008) 993–996.

Please cite this article in press as: L. Graening, B. Sendhoff, Shape mining: A holistic data mining approach for engineering design, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.002