Discontinuities in Continuous Scatterplots

Dirk J. Lehmann and Holger Theisel

Abstract—The concept of continuous scatterplot (CSP) is a modern visualization technique. The idea is to define a scalar density value based on the map between an n-dimensional spatial domain and an m-dimensional data domain, which describe the CSP space. Usually the data domain is two-dimensional to visually convey the underlying, density coded, data. In this paper we investigate kinds of map-based discontinuities, especially for the practical cases $n = m = 2$ and $n = 3 \mid m = 2$, and we depict relations between them and attributes of the resulting CSP itself. Additionally, we show that discontinuities build critical line structures, and we introduce algorithms to detect them. Further, we introduce a discontinuity-based visualization approach – called contribution map (CM) – which establishes a relationship between the CSP's data domain and the number of connected components in the spatial domain. We show that CMs enhance the CSP-based linking & brushing interaction. Finally, we apply our approaches to a number of synthetic as well as real data sets.

Index Terms—Discontinuity, Scatterplot, Topology, Data Visualization.

1 Introduction

To enhance the user's perceptual effectiveness and to ease the interpretation it is important to have smooth data visualizations in arbitrary degree of detail and without artifacts like gaps or bucklings. The continuous scatterplot (CSP) [2] is such a technique which enhances the classical concept of (discrete) scatterplots. Instead of scattered samples it visualizes a continuous density function based on a mapping function between an n-dimensional input domain and an m-dimensional output domain. Hence, CSPs allow zooming into regions of interest. In practice the application area of a CSP is mainly 2-3 dimensional time-independent data or – with additional time dimension – 3-4 dimensional time-dependent data. Such data frequently occur in nature or in computational science.

Due to the mathematical nature of CSP there has to be an underlying topological structure with discontinuities which is meaningful for the interpretation of the data. With respect to the case of a one-dimensional data domain ($m = 1$) Carr et al. [6] discussed some problems using histograms to describe an underlying function distribution. Our work investigates kinds of meaningful discontinuities for a two-dimensional data domain ($m = 2$), especially for the practically relevant time-independent case of 2-3 dimensional data. Informally, discontinuities are points, lines or areas where an event happens "suddenly" and not continuously. A more theoretical perspective on discontinuities is given e.g., in [8].

The main contributions of this paper are:

1) The introduction of two different kinds of discontinuities for CSPs of the case $n = m = 2$.
2) The analysis of relations between these discontinuities of the case $n = m = 2$ and the visual characteristics of CSPs themselves.
3) The description of applications for linking & brushing and – with the concept of contribution maps – a new visualization approach for the visualization of structural information between the data domain and the spatial domain.
4) The introduction of four different kinds of discontinuities for CSPs of the case $n = 3 \mid m = 2$.
5) The analysis of relations between these discontinuities of the case $n = 3 \mid m = 2$ and the visual characteristics of the CSPs themselves.

2 Related Work & Background

The multivariate data visualization approach of CSPs has been introduced by Bachentahler and Weiskopf [2] to enhance the classical concept of discrete scatterplots in a continuous manner. It addresses data sets which can be described by a continuous mapping τ of n independent variables onto m dependent variables: $\tau : \mathbb{R}^n \to \mathbb{R}^m$. The space $\mathbb{R}^n$ is called the *spatial domain* because the data are measured there. The space $\mathbb{R}^m$ is the *data domain*. In practice, the domains are mostly sampled discretely, and the continuous mapping τ might be obtained by a (multivariate) interpolation technique, like Shepard interpolation [11] or radial basis (RBF) interpolation [7].

The m-dimensional CSP corresponds to a scalar density function $\sigma(\xi = \tau(\mathbf{x}))$, whereas $\mathbf{x} \in \mathbb{R}^n$ is an element of the spatial domain and $\xi = \tau(\mathbf{x}) \in \mathbb{R}^m$ is the element mapped onto the data domain. The approach requires a scalar density description $s(\mathbf{x})$ for each spatial domain element $\mathbf{x}$ which is typically uniform, e.g., $s(\mathbf{x}) = 1$. Integrating the elements $s(\mathbf{x}) \in V$ of a spatial domain volume V results into the mass equation $m_{sd} = \int_V s(\mathbf{x})\, \mathbf{d}^n\mathbf{x}$. Further, the mass equation $m_{dd} = \int_{\phi=\tau(V)} \sigma(\xi)\, \mathbf{d}^m\xi$ for the data domain results from integrating the elements $\sigma(\xi = \tau(\mathbf{x})) \in \phi$ of the corresponding volume element $\phi = \tau(V)$ of the data domain. Assuming that the mapping τ fulfills the property of mass conservation $m_{sd} = m_{dd}$, a general implicit equation of the CSP's scalar density function $\sigma(\xi)$ is:

$$\int_{\Phi=\tau(V)} \sigma(\xi)\, \mathbf{d}^m\xi = \int_V s(\mathbf{x})\, \mathbf{d}^n\mathbf{x}. \tag{1}$$

This equation has already been solved for different cases: if $n = m$, an element ξ of the data domain is connected with an element $\mathbf{x} = \tau^{-1}(\xi)$ in the spatial domain. The implicit Equation (1) can be rewritten by using the transformation theorem for integrals with the Jacobian $\mathbf{D}(\tau)$:

$$\int_V \sigma(\xi = \tau(\mathbf{x}))\, |\det(\mathbf{D}(\tau))(\mathbf{x})|\, \mathbf{d}^n\mathbf{x} = \int_V s(\mathbf{x})\, \mathbf{d}^n\mathbf{x}. \tag{2}$$

By rearranging Equation (2), the explicit CSP density function for the $m = n$ case is obtained:

$$\sigma(\tau(\mathbf{x})) = \frac{s(\mathbf{x})}{|\det(\mathbf{D}(\tau))(\mathbf{x})|}. \tag{3}$$

If $n > m$, an element ξ of the data domain is connected to a contour $\mathbf{x}_c = \tau^{-1}(\xi)$ in the spatial domain with the dimension $n - m$. Thus, the determinant $\det(\mathbf{D}(\tau)(\mathbf{x}))$ is not defined anymore, and the transformation theorem does not apply. Therefore, Equation (3) has to be adapted as an integral of all elements $\mathbf{x}_t \in \mathbf{x}_c$ over that contour $\mathbf{x}_c$:

$$\sigma(\xi) = \int_{\tau^{-1}(\xi)} \frac{s(\mathbf{x}_t)}{|Vol(\mathbf{D}(\tau))(\mathbf{x}_t)|}\, \mathbf{d}^{(n-m)}\mathbf{x}_t. \tag{4}$$

- *The authors are with the Department of Simulation and Graphics at the University of Magdeburg, Germany, E-mail: {dirk,theisel}@isg.cs.uni-magdeburg.de*

Manuscript received 31 March 2010; accepted 1 August 2010; posted online 24 October 2010; mailed on 16 October 2010.
For information on obtaining reprints of this article, please send email to: tvcg@computer.org.

Note that a number k of spatial domain contours $\mathbf{x}_{c_i}$, $i = 1,\dots,k$, for the $n > m$ case as well as spatial domain elements $\mathbf{x}_i$, $i = 1,\dots,k$, for the $n = m$ case can contribute a density $\sigma(\xi = \tau(\mathbf{e}_i))$ to the same data domain element ξ (with $\mathbf{e}_i = \mathbf{x}_{c_i}$ for the $n > m$ case and $\mathbf{e}_i = \mathbf{x}_i$ for the $n = m$ case). Therefore, the total data domain density $\sigma(\xi)_{all}$ is the superposition of all these single density values:

$$\sigma(\xi)_{all} = \sum_i^k \sigma(\xi = \tau(\mathbf{e}_i)). \tag{5}$$

In [3] the same authors presented an approximation approach to render CSPs for arbitrary dimension numbers of the spatial domain as well as the data domain. Based on the implicit density function (Equation (1)) they describe an approximation as follows:

$$s(\mathbf{x})\; V \approx \sigma(\xi)\; \Phi \;\;\Leftrightarrow\;\; \sigma(\xi = \tau(\mathbf{x})) \approx \frac{s(\mathbf{x})\; V}{\Phi = \tau(V)}.$$

Due to the fact that the spatial domain density is mostly uniform ($s(\mathbf{x}) = 1$), the density $\sigma(\tau(\mathbf{x}))$ is finally defined as ratio between V and the mapping $\tau(V)$:

$$\sigma(\tau(\mathbf{x})) = \frac{V}{\tau(V)};\; \mathbf{x} \in V.$$

The approximation error is directly correlated with $\tau(V)$. Thus, to steer the error a threshold $\Phi_{\max} = \tau(V)_{max}$ is used as upper limit for the measure of the mapped volume. If the volume $\tau(V)$ is larger than the threshold $\Phi_{\max}$, an adaptive dividing step of V is recursively done.

To accelerate the rendering of CSPs, a GPU-based algorithm for input data on tetrahedral grids has been introduced in [1]. It combines a cache concept with a sorting strategy to master the typical computational overhead with parallel computing.

Further, Heinrich and Weiskopf [9] presented a concept for continuous parallel coordinates based on CSPs. They described a continuous transformation between two-dimensional CSPs and two-dimensional continuous parallel coordinates with the aid of the point-line-duality and the mass conservation.

3 Discontinuities of the Case $n = m = 2$

In this section we consider the case of a two-dimensional spatial domain and data domain. Such data often occurs in climate or geology research. Typical data sets for that case represent physical variables in the data domain, like temperature, (atmospheric) pressure or stress, with respect to a two-dimensional spatial domain for instance a seabed, terrains or material surfaces. Such data sets are either static data sets or single slices of a time-dependent one. To visualize all slices as a stack of CSPs could under certain circumstances be an appropriate visualization technique for the time-dependent case. However, to handle this case we will define different models of discontinuities in the next sections.

3.1 Discontinuity Model of the Mapping Function

Based on the density function (3) we are interested in discontinuities. A discontinuity is an element $\xi = \tau(\mathbf{x})$ of the data domain with an infinite density $\sigma(\xi)$. Such elements are explicitly described by the limit:

$$\sigma(\xi = \tau(\mathbf{x})) = \pm\infty \;\;\Leftrightarrow\;\; \lim_{\det(\mathbf{D}(\tau)(\mathbf{x}))\to\pm 0} \frac{s(\mathbf{x})}{|\det(\mathbf{D}(\tau)(\mathbf{x}))|}.$$

The limit leads to an appropriate criterion for a discontinuity element ξ_u:

$$\xi_u \;\;\Leftrightarrow\;\; \det(\mathbf{D}(\tau)(\mathbf{x})) = 0. \tag{6}$$

Note that the discontinuity ξ_u depends on the mapping function τ and is independent of the spatial domain density function $s(\mathbf{x})$.

3.1.1 Critical Curves

The union of adjacent discontinuities ξ_u forms coherent *critical curves* ξ_c within the data domain, given by $\xi_c = \bigcup \xi_u$. As shown in Equation (5), the density $\sigma(\xi)_{all}$ of a data domain element ξ consists of superpositioned scalar density values $\sigma(\tau(\mathbf{x}_i) = \xi), i = 1,\dots,k$. However, Equation (6) has shown that it is necessary to consider only single density contributions $\sigma(\tau(x_i) = \xi)$ and not a superposition of it. On the other hand, $\sigma(\xi)_{all}$ as well as single density contributions $\sigma(\tau(x_i) = \xi)$ will in practice be finite due to numerical reasons. Therefore, it is ambiguous to detect discontinuities directly from the density function.

In analogy to level-set methods the approach embeds the density and the mapping function into a higher dimensional function to ease the discontinuity detection. Thus, we propose a parametric surface $S_{\xi,d} : \mathbb{R}^2 \to \mathbb{R}^3$ where the first two components are the mapping function $\xi = \tau(\mathbf{x})$ and the last component d is the signed reciprocal of the corresponding density value $\sigma(\xi = \tau(\mathbf{x}))$:

$$S_{\xi,d} = \begin{pmatrix} \xi \\ d \end{pmatrix} = \begin{pmatrix} \xi_1 = \tau_1(x_1,x_2) \\ \xi_2 = \tau_2(x_1,x_2) \\ d = \det(\mathbf{D}(\tau)(\mathbf{x}))) \end{pmatrix}, \tag{7}$$

with

$$\xi = \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} \text{ and } \tau(\mathbf{x}) = \begin{pmatrix} \tau_1(x_1,x_2) \\ \tau_2(x_1,x_2) \end{pmatrix}.$$

With respect to Equation (6) and (7) we can give a more formal description now: critical curves correspond to those isolines of the surface $S_{\xi,d}$ which satisfy $S_{\xi,d=0}$. This shows that critical curves for non-degenerated mappings τ are coherent curves in the data domain which are given by:

$$\bigcup \xi_u \;\;\Leftrightarrow\;\; S_{\xi,d=0}. \tag{8}$$

3.1.2 Critical Curves Detection Algorithm

By sampling the parametric surface $S_{\xi,d}$, this surface can be represented by the resulting mesh $M_{\xi,d}$. The approximation error $||M_{\xi,d} - S_{\xi,d}||$ depends only on the sampling resolution and can be chosen as small as desired. Detecting critical curves now means to clip that mesh with the plane $d = 0$ by using standard clipping approaches. The resulting polylines are the critical curves.

3.2 Discontinuity Model of the Boundary

In this section we introduce another discontinuity type: *boundary curves*. The spatial domain is usually finite and corresponds to a two-dimensional rectangle. The boundary $\mathbf{b}$ consists of four boundary edges. As Figure 1 (left) illustrates, an element of the spatial domain boundary $\mathbf{x_b} \in \mathbf{b}$ is mapped onto an element $\xi_\mathbf{b} = \tau(\mathbf{x_b})$ of the data domain. Additionally, the element $\mathbf{x_b}$ is a boundary element of a corresponding infinitesimal volume V of the spatial domain, and the mapped element $\xi_\mathbf{b} = \tau(\mathbf{x_b})$ is also a boundary element of the mapped volume $\Omega = \tau(V)$.

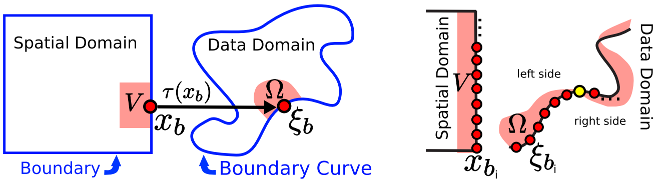

Fig. 1. Boundary curves: (left) A boundary element of the spatial domain gets to an element of the boundary curve in the data domain by mapping it. (right) The union of mapped boundary elements forms a boundary curve (implied by red points). The yellow point indicates where the corresponding volume (pink) changed from the left side of the boundary curve to the right side.

Figure 1 (right) and the last explanation depict the following: an arbitrary amount of adjacent boundary elements $\mathbf{x}_{b_i}, i = 1,\dots w$, being mapped and forming a boundary curve. The corresponding volume V being mapped and forming the volume Ω which is mainly on one side of the boundary curve. Therefore, boundary curves produce curves in the data domain where the density contribution from one side of the curve to the other side suddenly disappears. That is the reason why these curves form discontinuities.

Finally, Figure 2 shows critical and boundary curves for a CSP with $m = n = 2$. The underlying synthetic 2D scalar data sets are sampled from the function

$$\begin{pmatrix}\xi_1(\mathbf{x})\\ \xi_2(\mathbf{x})\end{pmatrix} = \begin{pmatrix}\sin(x_1)\,x_1 + \sin(x_2)\,x_2\\ \sin(x_1)\,x_2 + \sin(x_2)\,x_1\end{pmatrix}$$

on a uniform grid with cell size 0.1 within the spatial domain $[0,3] \times [0,3]$.

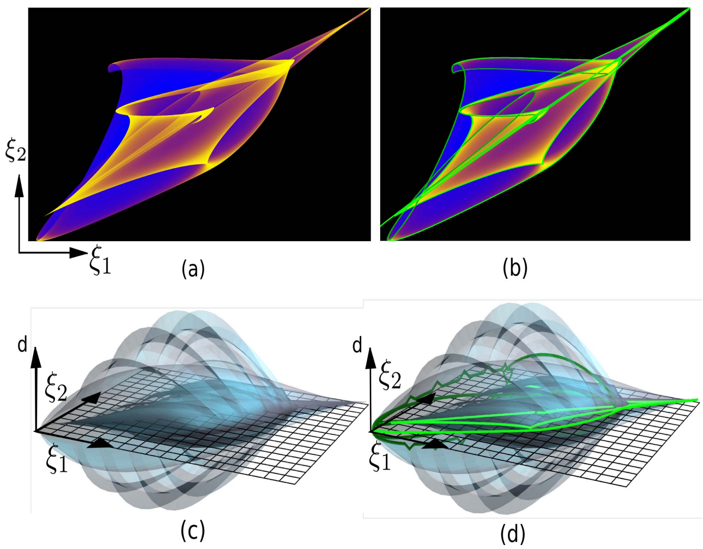

Fig. 2. Critical and boundary curves of a CSP of the case $m = n = 2$ based on a synthetic data set: (a) CSP of the data set with respect to [2], (b) Critical (green) and boundary curves (dark green) in the CSP, (c) According 3D mesh $M_{\xi,d}$ (blue) that approximates the surface $S_{\xi,d}$, (d) Mesh $M_{\xi,d}$ with critical (green) and boundary curves (dark green).

3.3 Analysis

In this section we investigate limitations and consequences of the concept of boundary and critical curves. Furthermore, we derive relations between these curves and visual characteristics of the CSP.

3.3.1 Limitations

The critical curves correspond to the null set where the mapping function (with respect to the density function) is not a diffeomorphism. If the function is additionally degenerated, e.g., piecewise constant, the critical curves become critical surfaces because the zero-crossings of the surface $S_{\xi,d=0}$ cover a whole region instead of just curves. Such critical areas are structurally unstable, i.e., they would break by adding noise. For our work we assume only structurally stable (smooth) mapping functions as input data.

3.3.2 Contributions of the Spatial Domain to the Data Domain

As already mentioned, a scalar density value $\sigma(\xi)_{all}$ of the data domain consists of a number k of different superpositioned density elements $\sigma(\tau(\mathbf{x}_i) = \xi)$, $i = 1,\ldots,k$. Therefore, we denote the number of contributing elements $\mathbf{x}_i$, $i = 1,\ldots,k$, the *contribution number* $k(\xi)$ of ξ. This number describes how many (isolated) elements of the spatial domain are used to produce the density of a certain element of the data domain. Formally, $k(\xi)$ is defined as:

$$k(\xi) = \text{ number of elements } \mathbf{x}_i \text{ with } \mathbf{x}_i = \tau^{-1}(\xi).$$

How can this contribution number be detected for each data domain element? Let us consider the parametric surface $\mathbf{S}_{\xi,d}$ which was introduced in Section 3.1.1. By sending a ray $\mathbf{r}(\xi)$ from ξ in the directions $\pm d$ the contribution number $k(\xi)$ is equal to the number of intersection points between $\mathbf{r}(\xi)$ and $\mathbf{S}_{\xi,d}$ (see Figure 3 (a)). The problem of detecting the contribution number is transformed into a ray casting problem.

Another property is also of interest: the contribution number is constant within the data domain until either (i) the surface $\mathbf{S}_{\xi,d}$ ends or (ii) the last component n_z of the surface normal $\mathbf{n} = \mathbf{n}_{\mathbf{S}_{\xi,\mathbf{d}}} = (n_x, n_y, n_z)^T$ changes its sign. Figure 3 (a) demonstrates this. In detail this means:
(i) The first kind of locations in the data domain where the contribution numbers change is where the surface $\mathbf{S}_{\xi,d}$ ends, because there the number of intersection points decreases by one.
(ii) The second kind of locations is where the n_z-component of the surface normal $\mathbf{n}_{\mathbf{S}_{\xi,\mathbf{d}}}$ is changing the sign, which is also depicted in Figure 3 (a). The normal $\mathbf{n}_{\mathbf{S}_{\xi,\mathbf{d}}}$ of a surface is defined as cross product over the gradients:

$$\mathbf{n}_{\mathbf{S}_{\xi,\mathbf{d}}} = \begin{pmatrix}n_x\\ n_y\\ n_z\end{pmatrix} = \begin{pmatrix}\xi_{1_{x_1}}\\ \xi_{2_{x_1}}\\ d_{x_1}\end{pmatrix} \times \begin{pmatrix}\xi_{1_{x2}}\\ \xi_{2_{x2}}\\ d_{x2}\end{pmatrix},$$

with the partial derivatives $\xi_{i_{x_j}}$, d_{x_j}, $i, j \in \{1,2\}$. With respect to the n_z-component of that normal $n_{\mathbf{S}_{\xi,d}}$ this equation leads to:

$$n_z = \xi_{1_{x_1}}\,\xi_{2_{x_2}} - \xi_{2_{x_1}}\,\xi_{1_{x_2}} = \det(\mathbf{D}(\tau)(\mathbf{x})).$$

This shows that the n_z-component of the surface normal $\mathbf{n}_{\mathbf{S}_{\xi,\mathbf{d}}}$ is equivalent to the d-component of the surface $\mathbf{S}_{\xi,d}$ itself. Hence, the n_z-component of the surface normal $\mathbf{n}_{\mathbf{S}_{\xi,\mathbf{d}}}$ and so the d-component of the surface $\mathbf{S}_{\xi,d}$ change their sign at $d = \det(\mathbf{D}(\xi)) = n_z = 0$ (see Figure 3 (b)). With respect to Equation (8) critical curves are defined by $\det(\mathbf{D}(\xi)) = 0$ and thus they also have to be boundaries of areas of constant contribution numbers in the data domain. It follows that $k(\xi)$ is a piecewise constant function.

Note that self-intersections of the surface $\mathbf{S}_{\xi,d}$ do not need to be considered for the question where the contribution number changes because they do not influence the number of intersection points.

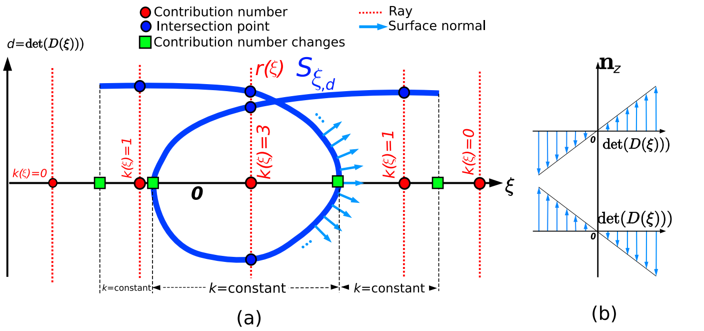

Fig. 3. (a) Detection of the contribution number $k(\xi)$ from the surface $\mathbf{S}_{\xi,d}$ with ray cast: Contribution numbers are piecewise constant and change their value depending on attributes of the surface normals and the surface boundary. (b) Relation between surface normal component n_z and component $d = \det(\xi)$ of that surface. Both change their signs synchronously.

3.3.3 Relations between Critical/Boundary Curves and CSPs

Figure 4 (a-b) shows a comparison of the edges of a CSP (a) with boundary and critical curves (b). It can be noticed that edges/discontinuities in the image space of that CSP relate to critical as well as boundary curves of the case $n = m = 2$. Boundary and critical curves are structures where either a density suddenly ends or gets infinite. On the other hand, it has already been shown that boundary and critical curves also correspond to boundaries of constant contribution numbers. The contribution number changes by one when crossing a boundary curve and by two when crossing a critical curve, as shown in Figure 3. Consequently, the contribution number boundaries also relate to edges in the image space of the CSP. Concluding, Figure 4 (c) illustrates that isolines of the surface $\mathbf{S}_{\xi,d}$ with $d \neq 0$ are not edges in the CSP.

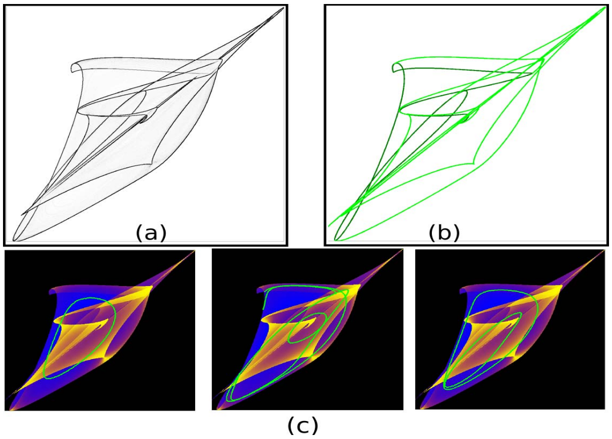

Fig. 4. Relation between CSPs and critical and boundary curves: (a) Edges in a CSP detected with a canny [5] image process operation, (b) Critical and boundary curves of the same CSP detected with the described approaches, (c) Counterexample: Isolines (green) of surface $\mathbf{S}_{\xi,d}$ with $d \neq 0$ are neither critical curves nor image edges (left to right) d=3, d=5, d=7.

3.4 Applications

In this section we present applications and demonstrate real world examples for discontinuities of the discussed $m = n = 2$ case.

3.4.1 Discontinuities and Linking & Brushing

Linking & brushing (L&B) is an important interaction task of explorative visual search. This technique has been introduced by Becker and Cleveland [4] and is used to select exactly such elements from the spatial domain and/or the data domain which are in a certain relation to each other. Usually, these elements will be highlighted in both domains to give a visual feedback to the user. With L&B techniques the user is often able to find hidden patterns and relations within a multivariate data set.

Two L&B directions are possible for CSPs: from the spatial domain to the data domain and vice versa. Selecting a volume V of the spatial domain links to a volume $\Omega = \tau(V)$ of the data domain. In contrast, selecting the same volume Ω in the data domain links to a number of different volumes $V_i = \tau^{-1}(\Omega)$, $i = 1,\ldots,f$, in the spatial domain: this last case is because different elements of the volume Ω consist of different elements of the spatial domain. The effect has already been discussed in Section 3.3.2. Thus, the resulting spatial domain volume V_{res} of the data domain area Ω is:

$$V_{res} = \bigcup_{b=1}^{f} (V_b \subseteq \tau(\Omega)^{-1}) \text{ with } V \subseteq V_{res}.$$

If all elements $\xi \in \Omega$ of the data domain volume Ω have the same contribution number $k(\xi)$, the corresponding elements in the spatial domain form at most k different volumes, or less due to volume fusions in the spatial domain: $f \leq k$.

If the elements $\xi \in \Omega$ are a mixture of different contribution numbers, the upper limit f of different volumes depends on the behavior of the mapping function. However, we already know that such a volume Ω that contains a mixture of different contribution numbers has to cross either a boundary or a critical curve because these curves are boundaries of areas of constant contribution numbers. Nevertheless, the area V and Ω link in both linking directions to each other.

Now, we propose a new visualization technique called *contribution map* (CM) to improve the user's ability to recognize relations between the data and the spatial domain for the complex case of linking from the data to the spatial domain. This technique is derived from the contribution number concept which has been discussed in Section 3.3. A CM is defined as the color-coded visualization of all contribution numbers $k(\xi)$ of the elements ξ of that data domain. Based on the results of the discontinuity analysis in Section 3.3.3 it is obvious that the image edges of a CM correspond on the one hand with the boundary and critical curves, and on the other hand with the edges of that CSP. Figure 5 shows a CM.

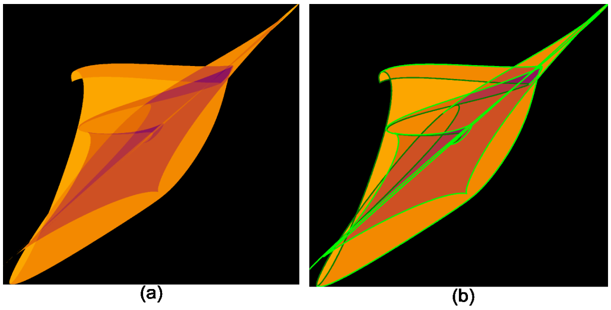

Fig. 5. Contribution Map: (a) Color-coded contribution map of a CSP: the more red, the larger is the contribution number. (b) The same contribution map with critical (green) and boundary curves (dark green).

Supported by a CM the user is able to visually estimate which elements of the data domain consist of more or less contribution of the spatial domain compared to other elements. In other words: the CM visualizes which areas of the data domain incorporate more or less areas of the spatial domain. Thus, a CM is a new source for structural information about the underlying data set.

Therefore, the CM visualization might be combined with an interactive L&B technique to introduce a new visual exploration tool for CSPs. So the user can select an area of interest in the data domain with a variable contribution number and, by L&B, the user can investigate the distribution behavior of the corresponding areas in the spatial domain to search a meaningful distribution pattern. Further, it is possible to explore relations between the visually coded variables of the data domain and the number of contributions (or connected components, respectively) in the spatial domain. Figure 6 illustrates an example of this exploration tool. The larger the contribution number $k(\xi)$ of the selected area in the data domain is, the larger is the corresponding linked area V_{res} in the spatial domain.

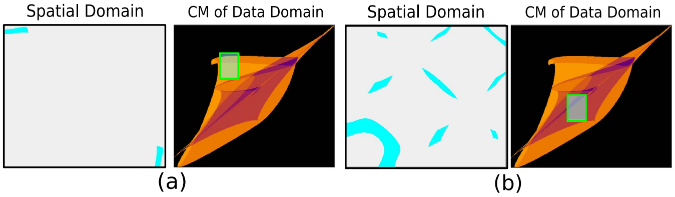

Fig. 6. Contribution Map combined with Linking & Brushing: The more red colored an element of the CM is, the larger is the underlying contribution number. (a) A selected area in the data domain (green bordered rectangle) with low contribution number (orange) links only to a small area in the spatial domain (cyan), (b) A selected area in the data domain (green bordered rectangle) with a higher contribution number (red) as before links to a larger area in the spatial domain (cyan) than before.

3.4.2 Alternative Linking & Brushing Concepts

To facilitate the intuitive understanding from which spatial domain element a data domain discontinuity element comes from, an animation

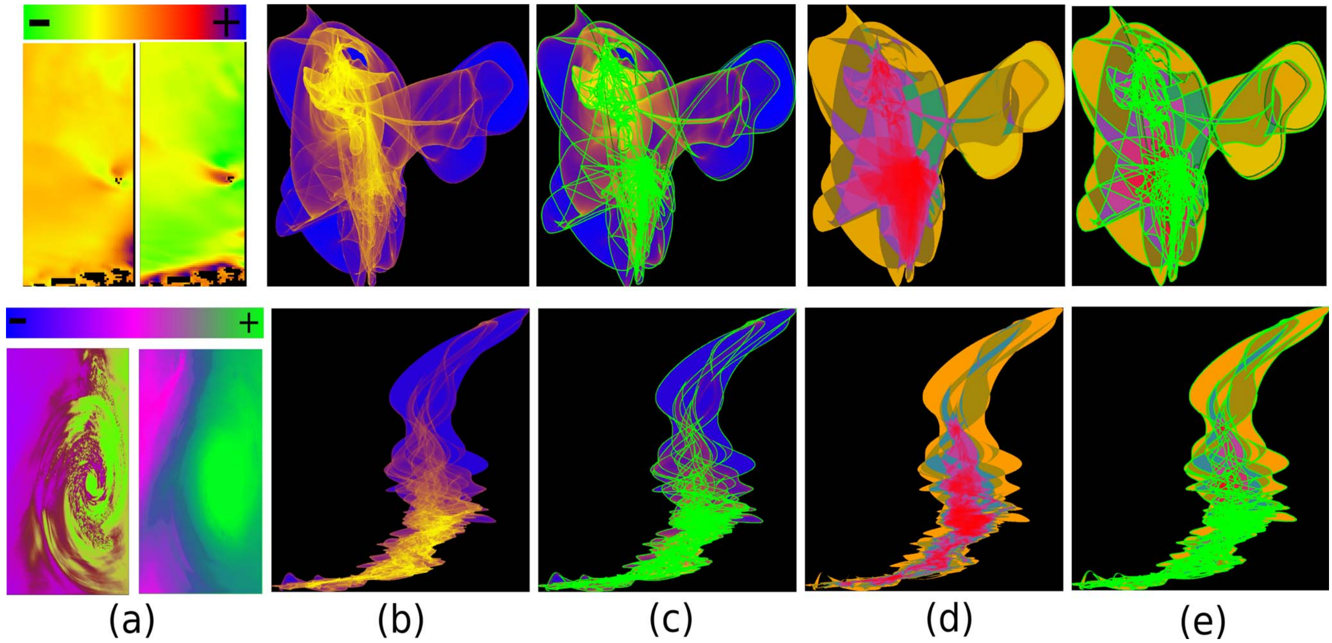

Fig. 7. Discontinuity application "north sea" (up) and "Hurricane Isabel" (down) data set: (a up) The underlying 2D scalar fields of the CSP calculations with a grid resolution of 130×30, color coded x velocity component and y velocity component of the sea current. (a down) The underlying 2D scalar fields of the CSP calculations with a grid resolution of 500×500, color coded distribution of the pressure and of the temperature. (b) The resulting CSPs. (c) The CSPs with critical (green) and boundary curves (dark green). (d) The corresponding CMs. (e) The CMs with critical lines (green) and boundary curves (dark green)

showing how the spatial domain morphs into the data domain can be used. The advantage of the concept is that an animation is a natural way to visualize relations because in everyday life the user is surrounded by a lot of animations and therefore accustomed to interpret them. Some of these animation aspects for visualizations have been investigated by Tversky et al. [14].

The animation process from the spatial to the data domain is realized as follows: the spatial domain is covered by a (e.g., regular) grid G. Each point $\mathbf{x}_g = (\mathbf{x}_{g_1}, \mathbf{x}_{g_2}, 0)^T \in G$ corresponds to a data domain point $\xi_g = (\xi_{g_1} = \tau_1(\mathbf{x}_g), \xi_{g_2} = \tau_2(\mathbf{x}_g), d = \det(\mathbf{D}(\tau)(\mathbf{x}_g))^T \in M_{\xi,d}$ of a mesh $M_{\xi,d}$. Note that this mesh is equivalent to the mesh $M_{\xi,d}$ introduced in Section 3.1.2. The animation is defined by moving each point $\mathbf{x}_g$ of the data domain linearly to the corresponding point ξ_g of the spatial domain. In other words: the backwards running animation unfolds the mesh – which represents the data domain – over the spatial domain and vica versa. All points whose last coordinate is zero over the whole animation are zero-crossings of that mesh $M_{\xi,d}$ and consequently discontinuities, as explained in Section 3.1.2.

To visually enhance this method, any mesh vertex should be colored with the corresponding density contributions, e.g., as vertex color. Further, an alpha blending might be used to avoid confusions of the user due to visual occlusions. Figure 8 presents a sequence of images of such an animation.

Finally, to complete the L&B discussion a last linking tool should be mentioned. Instead of linking manually selected areas in both domains to each other we can use a density threshold σ_{max} to link elements automatically: all elements $\mathbf{x}$ in the spatial domain and all elements ξ in the data domain can be selected and highlighted, which fulfills:

$$(\sigma(\tau(\mathbf{x})) > \sigma_{max}) \quad \wedge \quad (\sigma(\xi)_{all} \leq \sigma_{max}).$$

The intention is that no element $\mathbf{x}$ of the spatial domain can contribute a density value $\sigma(\tau(\mathbf{x}))$ to any element ξ of the data domain whose total density value $\sigma_{all}(\xi)$ is smaller than this contribution. That is due to the superposition characteristics mentioned in Equation (5). Hence, the L&B relation for this case is that only elements in both domains are selected that do not map to each other.

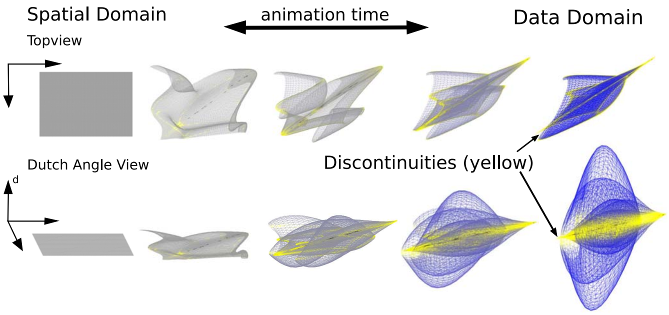

Fig. 8. Observing the evolution of discontinuities between spatial and data domain by use of animations. (left-right) An image strip of different animation time steps is presented for a top view (up) as well as for a dutch angle view (down). The discontinuities and zero-crossings, respectively are shown in yellow and the mesh is visualized as wire frame.

3.4.3 Discontinuities in Data Sets

We apply the approach described above to the "north sea" data set and to the "Hurricane Isabel" data set. The "north sea" data set contains flow information of the North Sea in the German Bight on 10/17/2008. We used the topmost 2D slice of that 3D flow because that data set is dominated by its horizontal components and can be considered as 2D vector field. The "Hurricane Isabel" data set is a model of a tropical hurricane (09/06/2003-09/19/2003 North Carolina) which contains 3D time-dependent information about different physical measurements, like atmospheric pressure, cloud formation or temperature. We used the variables pressure and temperature of time step 10 and slice 10 with a grid resolution of 500×500.

Figure 7 illustrates the resulting CSPs, critical and boundary curves as well as a the CMs for both data sets. The CSPs have been built on an Intel Core 2 Quad CPU (using a single core) with Linux OS and 3.5 GB RAM with the original approach for $n = m = 2$ CSPs of [2]. The interpolation has been done by RBFs of Gaussian type. The additional computation time for the surface $\mathbf{S}_{\xi,d}$ is negligible because the surface is a by-product of the CSP calculations. The computation time of the critical and boundary curves did not exceed a half second in our examples.

4 Discontinuities of Case $n = 3 \mid m = 2$

In this section we consider the case of a three-dimensional spatial domain and a two-dimensional data domain. That special case of $n > m$ is especially important for data sets of flow which often occur, e.g., in oceanography, plasma or aerodynamic physics.

4.1 Discontinuity Model of the Mapping Function

Due to the different dimensionality of the spatial domain and the data domain the discontinuities of this CSP case have a more complicated character as such described in Section 3.1.

Let us consider an element $\xi = (\xi_1, \xi_2)^T$of the data domain: the component ξ_1 is represented by an isosurface $\mathbf{S}_{\xi_1} = \tau^{-1}(\xi_1)$ and the component ξ_2 is represented by an isosurface $\mathbf{S}_{\xi_2} = \tau^{-1}(\xi_2)$ in the spatial domain. Thus, the data domain element ξ in the spatial domain is represented by an intersection curve $I_\xi = \mathbf{S}_{\xi_1} \bigcap \mathbf{S}_{\xi_2}$. Figure 9 (a) demonstrates that the intersection curve is a (one-dimensional) isoline of a constant ξ and is embedded in the three-dimensional spatial domain.

Unfortunately, no closed parametric definition of an isoline I_ξ is possible. Therefore, we propose to construct a vector field $\mathbf{q}(\mathbf{x})$ over the spatial domain whose stream lines correspond exactly to the isolines $I_{\xi=\tau(\mathbf{x})}$: let $\nabla\tau_1(\mathbf{x})$ and $\nabla\tau_2(\mathbf{x})$ be the gradients of the mapping $\tau(\mathbf{x}) = (\tau_1(\mathbf{x}), \tau_2(\mathbf{x}))^T$ which point in the direction of maximal slope of $\xi_1 = \tau_1(\mathbf{x})$ and $\xi_2 = \tau_2(\mathbf{x})$, respectively. Then the cross product $\mathbf{q}(\mathbf{x}) = \nabla\tau_1(\mathbf{x}) \times \nabla\tau_2(\mathbf{x})$ is the vector which points into the direction where the slope of the corresponding domain element $\xi = (\xi_1, \xi_2)^T$ has to be zero. This means, that the vector $\mathbf{q}(\mathbf{x})$ points in the direction of the isolines pathway. Thus, the 3D vector field

$$\mathbf{q}(\mathbf{x}) = \nabla\tau_1(x) \times \nabla\tau_2(x) = \nabla\xi_1 \times \nabla\xi_2 = \begin{pmatrix} \xi_{1_{x_2}}\,\xi_{2_{x_3}} - \xi_{1_{x_3}}\,\xi_{2_{x_2}} \\ \xi_{1_{x_3}}\,\xi_{2_{x_1}} - \xi_{1_{x_1}}\,\xi_{2_{x_3}} \\ \xi_{1_{x_1}}\,\xi_{2_{x_2}} - \xi_{1_{x_2}}\,\xi_{2_{x_1}} \end{pmatrix}, \quad (9)$$

with the partial derivations $\xi_{i_{x_j}}$, $i, j \in \{1,2\}$, is the desired vector field where the stream lines correspond to the isolines $I_{\xi=\tau(\mathbf{x})}$. Figure 9 (b-c) illustrates that.

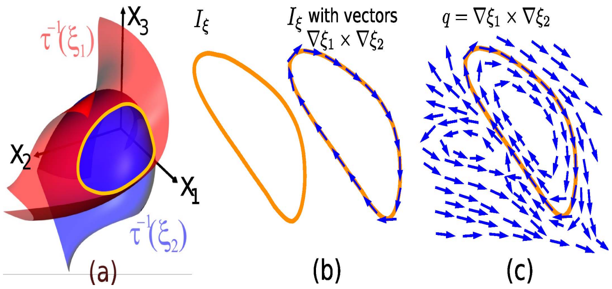
$\tau^{-1}(\xi_1)$
X_3
X_2
X_1
$\tau^{-1}(\xi_2)$
(a)
I_ξ
(b)
I_ξ with vectors $\nabla\xi_1 \times \nabla\xi_2$
$q = \nabla\xi_1 \times \nabla\xi_2$
(c)

Fig. 9. From isosurfaces to the vector field: (a) The intersection of isosurfaces $\mathbf{S}_{\xi_1} = \tau^{-1}(\xi_2)$ (red) and $\mathbf{S}_{\xi_2} = \tau^{-1}(\xi_2)$ (blue) yields the intersection curve (orange) which represents the isoline $\xi = (\xi_1, \xi_2)^T$ in the spatial domain. (b) The isoline (left, orange) with blue colored tangential vectors of $\mathbf{q}(\mathbf{x})$ (right). (c) Blue colored 3D vector field $\mathbf{q}$ which contains the stream lines of corresponding isolines.

The integration time t_{I_ξ} of a stream line in $\mathbf{q}(\mathbf{x})$ is infinite or disappears at the same location $\mathbf{x}$ as the density contribution $\sigma(I_\xi)$ of the corresponding isolines would do (see Equation (4)). Thus, we can use the integration time t_{I_ξ} within the vector field $\mathbf{q}$ to define different cases of discontinuities.

Center Discontinuity Figure 10 (a) illustrates that a closed isoline I_ξ in the spatial domain might shrink to an isolated point while varying ξ in the data domain. For that case the integration time t_{I_ξ} of the corresponding stream line is constant in a first order approximation. But when the isoline disappears, the integration time t_{I_ξ} as well as the corresponding density value $\sigma(I_\xi)$ gets suddenly from a constant value to zero (see Figure 10 (b)). We denote the isolated point as *center discontinuity* because of its center behavior in the vector field $\mathbf{q}(\mathbf{x})$: $||\mathbf{q}(\mathbf{x})|| = 0 \wedge Re(\lambda_1) = Re(\lambda_2) = 0$ where λ_i are the eigenvalues of the Jacobian $D(\mathbf{q})$ [15].

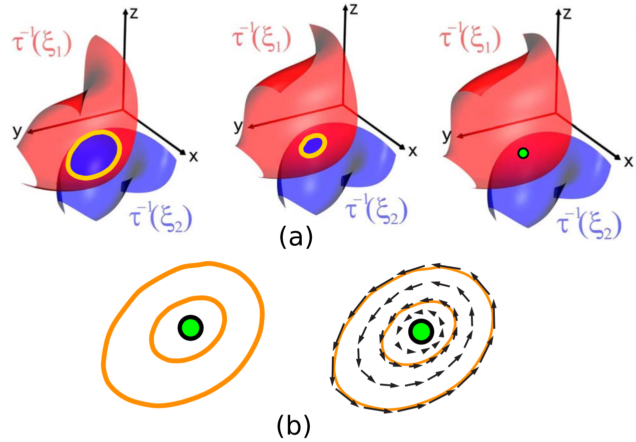

Fig. 10. Center Discontinuity: (a) Isoline (orange) shrinks to an isolated point while varying the isosurfaces of ξ_1 (red) and ξ_2 (blue): the isolated point where the isoline disappears is the center discontinuity (green). (b) Illustration of that isoline evolution (right) and the corresponding vector field $\mathbf{q}$ colored in blue (left).

Saddle Discontinuity As Figure 11 (a) shows, a closed isoline I_ξ in the spatial domain might split while varying ξ in the data domain. For that case, the integration time t_{I_ξ} of the corresponding stream line gets infinite as well as the corresponding CSP density value $\sigma(I_\xi)$. Figure 11 (c) depicts this. We denote the split point as a *saddle discontinuity* because of its saddle behavior in the vector field $\mathbf{q}$: $||\mathbf{q}|| = 0 \wedge Re(\lambda_1) \leq 0 \leq Re(\lambda_2)$.

Furthermore, the isoline which belongs to a saddle discontinuity separates two center discontinuities, as Figure 11 (c) depicts.

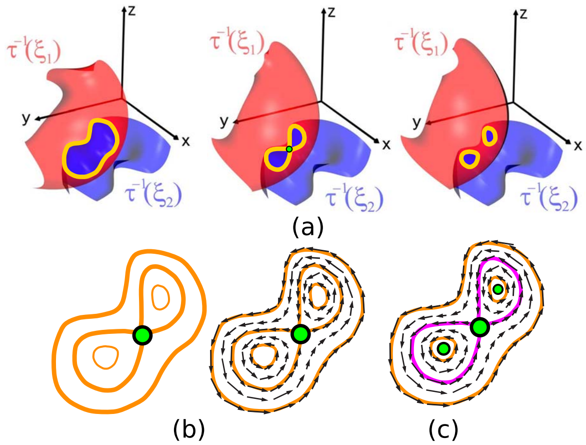

Fig. 11. Saddle Discontinuity: (a) An isoline (orange) splits while varying the isosurfaces of ξ_1 (red) and ξ_2 (blue): The split point is a saddle discontinuity (green). (b) Illustration of that isoline evolution (right) in contrast to the corresponding vector field $\mathbf{q}$ colored in blue (left). (c) Isoline of a saddle discontinuity (purple) separates two center discontinuities (additional green points).

4.1.1 Critical Curves

The set of the center as well as the saddle discontinuities form closed lines (unless they intersect the boundary) within the vector field $\mathbf{q}$, so-called feature lines [12, 15]. Therefore, we can define critical curves of a CSP as the feature lines of $\mathbf{q}$ that will be mapped from the spatial domain onto the data domain (Figure 12).

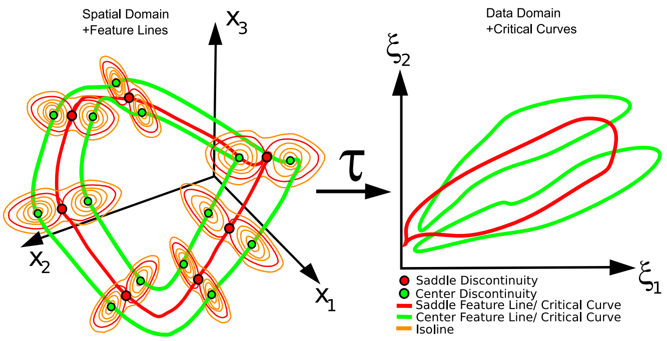

Fig. 12. Critical Curves: (left) a center feature line (green) is the union of adjacent center discontinuities (green points); a saddle feature line (red) is the union of adjacent saddle discontinuities (red points); (right) mapped feature lines form critical curves in the data domain; isolines are orange colored.

4.1.2 Detection of Critical Curves

To detect the critical curves we have to detect the feature lines of the vector field $\mathbf{q}$ first. That feature lines are structures of zero points in $\mathbf{q}$ with $\mathbf{q}(x) = \mathbf{0}$, as already seen in Section 4.1. From Equation (9) it follows that the zeros of $\mathbf{q}(\mathbf{x})$ correspond to the locations where the gradients of $\xi = \tau(\mathbf{x})$ are parallel:

$$\mathbf{q} = \mathbf{0} \quad \Leftrightarrow \quad \nabla\xi_1 || \nabla\xi_2.$$

Detecting parallel vectors is a well-discussed issue by Peikert and Roth [10]. Based on that, Theisel et al. [13] proposed a concept of feature flow field (FFF). A FFF $\mathbf{f}(\mathbf{q})$ of such a 3D vector field $\mathbf{q}$ is also a 3D vector field in which a feature line of $\mathbf{q}(\mathbf{x})$ corresponds to a stream line of $\mathbf{f}(\mathbf{q}(\mathbf{x}))$.

Detecting the feature lines of $\mathbf{q}$ is now an integration task in $\mathbf{f}$ starting at $\mathbf{f}(\mathbf{q}(\mathbf{x}) = \mathbf{0})$. In practice, this integration task in an FFF is numerically unstable. That is because on the one hand we are mostly not able to start exactly at a point $\mathbf{f}(\mathbf{q}(\mathbf{x}) = \mathbf{0})$, on the other hand the integration would lead away from the feature line due to the intrinsic integration error of the standard integrators. Therefore, Weinkauf et al. [15] discussed a concept of stable FFF (SFFF) that solves this stability problem: the approach enriches the environment of a stream line in $\mathbf{f}$ (which represents a feature line in $\mathbf{q}$) with an attracting property. In other words: the feature lines attract (massless) integration particles onto them. An integration that starts close to $\mathbf{f}(\mathbf{q}(\mathbf{x}) = \mathbf{0})$ is successful anyhow. This is illustrated in Figure 13 (a-b). SFFFs are defined (among other) for 3D vector fields that are the result of a cross product. The vector field $\mathbf{q}(\mathbf{x})$ is such a field (see Equation (9)) and therefore we can use a SFFF $\mathbf{h}(\mathbf{q}(\mathbf{x}))$ to detect the feature lines of $\mathbf{q}(\mathbf{x})$ by stable integration. The SFFF $\mathbf{h}(\mathbf{q}(\mathbf{x}))$ is defined as a weighted blending between the FFF $\mathbf{f}(\mathbf{q}(\mathbf{x}))$ and the attracting vector field $\mathbf{g}(\mathbf{q}(\mathbf{x}))$ as follows:

$$\mathbf{h}(\mathbf{q}(\mathbf{x})) = \mathbf{f}(\mathbf{q}(\mathbf{x})) + \frac{\alpha\, \mathbf{g}(\mathbf{q}(\mathbf{x}))}{||\mathbf{f}(\mathbf{q}(\mathbf{x}))||},$$

with

$$\mathbf{f}(\mathbf{q}(\mathbf{x})) = \begin{pmatrix} \det(\mathbf{q}_y, \mathbf{q}_z, \mathbf{a}) \\ \det(\mathbf{q}_z, \mathbf{q}_x, \mathbf{a}) \\ \det(\mathbf{q}_x, \mathbf{q}_y, \mathbf{a}) \end{pmatrix} \text{ with } \mathbf{a} = \begin{cases} \nabla\xi_1 & \text{if } ||\nabla\xi_1|| \geq ||\nabla\xi_2|| \\ \nabla\xi_2 & \text{else} \end{cases}$$

and

$$\mathbf{g}(\mathbf{q}(\mathbf{x})) = \begin{pmatrix} \det(\mathbf{q}_z, \mathbf{q}_x, \mathbf{a})\det(\mathbf{q}, \mathbf{q}_z, \mathbf{a}) - \det(\mathbf{q}_x, \mathbf{q}_y, \mathbf{a})\det(\mathbf{q}, \mathbf{q}_y, \mathbf{a}) \\ \det(\mathbf{q}_x, \mathbf{q}_y, \mathbf{a})\det(\mathbf{q}, \mathbf{q}_x, \mathbf{a}) - \det(\mathbf{q}_y, \mathbf{q}_z, \mathbf{a})\det(\mathbf{q}, \mathbf{q}_z, \mathbf{a}) \\ \det(\mathbf{q}_y, \mathbf{q}_z, \mathbf{a})\det(\mathbf{q}, \mathbf{q}_z, \mathbf{a}) - \det(\mathbf{q}_z, \mathbf{q}_x, \mathbf{a})\det(\mathbf{q}, \mathbf{q}_x, \mathbf{a}) \end{pmatrix}.$$

The $\mathbf{q}_i$ are the partial derivatives of the i'th component of the vector field $\mathbf{q}(\mathbf{x})$ and the parameter α steers the strength of the described attracting effect.

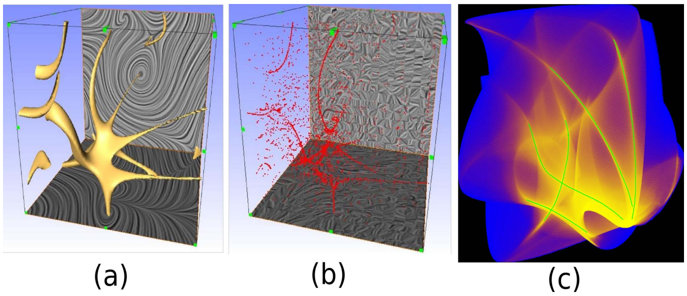

Fig. 13. Critical Curves for $m = 3 \mid n = 2$ case: (a) Level-sets of magnitude close to zero of a vector field $\mathbf{q}$. It contains all feature lines of $\mathbf{q}$. (b) A screenshot of one time step of a particle integration flow animation in the SFFF $\mathbf{h}(\mathbf{q})$ with 5000 randomly seeded particles (red). It can be seen that the particles are attracted to the feature lines and reproduce the structure of that level-set. (c) Critical curves (green) in the corresponding CSP for the vector field $\mathbf{q}$ of sub image (a). The characteristics of the level-set structures can still be recognized in the structure of the critical curves.

4.1.3 Critical Curves Detection Algorithm

We will give a brief overview of the critical curves detection algorithm.

1) Seedpoints: Boundary points with $\mathbf{q}(\mathbf{x}) = \mathbf{0}$ are seedpoints of feature lines that intersect the boundary. Closed feature lines become feature lines that intersect a boundary by recursively dividing the vector field $\mathbf{q}(\mathbf{x})$ with an octree scheme.

2) Integration: Forward and backward integration within the SFFF $\mathbf{h}(\mathbf{q}(\mathbf{x}))$ starting at the seedpoints delivers the feature lines $\mathbf{f}_l$, $l = 1, ..., s$.

3) Mapping: Generates critical curves of the CSP by mapping the feature lines $\mathbf{f}_l$ from the spatial domain to the data domain.

Furthermore, the critical curves can be classified in center or saddle critical curves by using the Jacobian analysis mentioned in Section 4.1. Figure 13 (c) shows an example of critical curves for a data set which is described in Section 4.4.

4.2 Discontinuity Model of the Boundary

The boundary box of the 3D spatial domain is also a source for different discontinuities in the data domain, so-called boundary curves. Figure 14 depicts two cases of such boundary curves that are explained in the following.

Box Edge Boundary Curve By mapping the edges of the box that bounds the 3D spatial domain onto the data domain, the *box edge boundary curves* arise in the data domain. These curves are comparable to the 2D boundary curves introduced in Section 3.2. The box edge boundary curves are curves in the data domain where the density from one side of the curve to the other side might suddenly disappear.

Boundary Switch Boundary Curve The stream lines of $\mathbf{q}(\mathbf{x})$ run either out of the boundary box or are completely embedded into it. Thus, there are stream lines of $\mathbf{q}(\mathbf{x})$ which hit the boundary in exactly one boundary switch point. These boundary switch points partition the boundary into segments, the so-called boundary switch curves [16]. They divide the boundary into inflow and outflow areas. Inflow areas

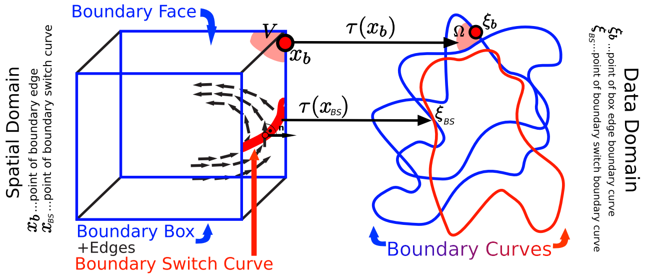
Boundary Face
V
x_b
$\tau(x_b)$
Ω
ξ_b
$\tau(x_{BS})$
ξ_{BS}
n
Spatial Domain
x_b ...point of boundary edge
x_{BS} ...point of boundary switch curve
Boundary Box
+Edges
Boundary Switch Curve
Boundary Curves
Data Domain
ξ_b ...point of box edge boundary curve
ξ_{BS}...point of boundary switch boundary curve

Fig. 14. Boundary curves in the data domain generated by mapping the box edges and the boundary switch curves onto the data domain.

are areas on the boundary where the stream lines flow into the boundary box and at outflow areas they flow out of the box. The behavior of outflow and inflow changes suddenly at the boundary switch curves (BSC) and therefore it is a discontinuity structure (see Figure 15 (left)). Further, the BSCs starts (or ends) on a box edge or are closed and they are free of self-intersections, as Figure 15 (right) depicts.

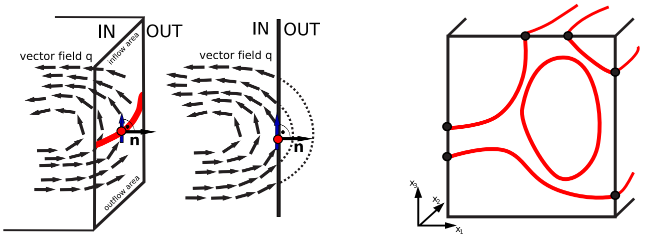

Fig. 15. Boundary Switch Curves: (left) Boundary switch points of stream lines of $\mathbf{q}(\mathbf{x})$ (blue arrows) form a boundary switch curve (red). (right) Boundary switch curves (red) on a boundary face either start/end (blue points) on a box edge or they are closed.

4.2.1 Boundary Switch Boundary Curve Detection

Within a BSC a vector of the vector field $\mathbf{q}(\mathbf{x})$ is parallel to the corresponding boundary face. Let $\mathbf{n}$ be the normal of a boundary face, then a BSC on that face is defined as

$$\mathbf{q}(\mathbf{x})\ \mathbf{n} = 0.$$

That equation defined a 2D scalar field $b(\gamma)$ for a given 2D boundary face because two of the three components of $\mathbf{q}(\mathbf{x})$ drop out. Here, $\gamma = (\gamma_1, \gamma_2)^T$ is an element of the local 2D coordinate system of that boundary face. A BSC is a zero isoline of that scalar field $b(\gamma)$ and can be detected as an integration over the stream lines of the corresponding co-gradient $(-\frac{\partial b}{\partial \gamma_2}, \frac{\partial b}{\partial \gamma_1})^T$, starting at points with $b(\gamma) = 0$.

4.2.2 Boundary Switch Boundary Curves Detection Algorithm

We will give a brief overview of the boundary switch boundary curve detection algorithm.

1) Seedpoints: Boundary edge points with $b(\gamma) = 0$ are seedpoints of a BSC intersecting the boundary edges. Closed BSCs become BSCs that intersect a boundary edge by recursively dividing the boundary faces with a quadtree scheme.

2) Integration: Forward and backward integration within the co-gradient $(-\frac{\partial b}{\partial \gamma_2}, \frac{\partial b}{\partial \gamma_1})^T$ starting at the seedpoints delivers the BSCs $\mathbf{b_l}$, $l = 1, ..., s$.

3) Mapping: Generates boundary switch boundary curves of the CSP by mapping the BSCs $\mathbf{b_l}$ from the spatial domain to the data domain.

4.3 Analysis

In this section we investigate the limitations and consequences of critical as well as boundary curves of the case $n = 3 \mid m = 2$ with respect to the CSP.

4.3.1 Limitations

For degenerated input data the critical curves may become critical surfaces similar to the $n = m = 2$ case as already discussed in Section 3.3.1. We assume structurally stable (smooth) mapping functions as input data.

Additionally, the detection algorithms of Section 4.1.3 and 4.2.2 are only able to detect those closed feature lines as well as closed BSCs which intersect the boundary and the boundary edges, respectively one time during the recursion (otherwise no seedpoints are detected). Furthermore, the standard integrators produce an integration error. In contrast to that, other numerical effects are negligible. Thus, we use the SFFF concept to weaken numerical errors and to get a stable integration.

4.3.2 Contributions of the Spatial Domain to the Data Domain

Due to the superposition characteristic mentioned in Equation (5) there are a number of isolines I_{ξ_i}; $i = 1, \ldots, k$ in the spatial domain that map to the same data domain element $\xi = \tau(I_{\xi_i})$. The number k of these isolines is the extension of the concept of contribution number introduced in Section 3.3 for the $n = 3 \mid m = 2$ case. The following question arises: are we able to detect the contribution number $k(\xi)$ for a chosen data domain element ξ? Unfortunately, in general we are not. This is due to the fact that the BSCs have demonstrated that an isoline does not need to be closed but might be split in the spatial domain. If we consider some split isoline parts in the spatial domain, we are not able to decide whether they belong to one isoline or to different isolines. Thus, we are not able to count the contribution number $k(\xi)$ (see Figure 16). Therefore, it is ambiguous how to define a contribution map for that $n = 3 \mid m = 2$ case. But even if we cannot detect the value of the contribution number, the following statements about its monotone behavior in the data domain are possible, with respect to Section 4.1 and 4.2.

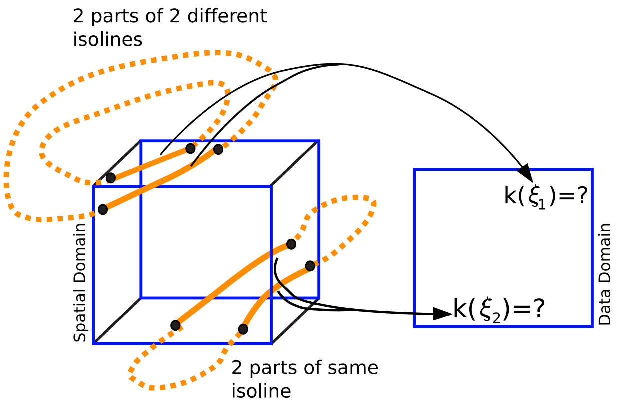

Fig. 16. Contribution Number Indetermination Issue: The contribution number $k(\xi)$ of a split isoline (orange) cannot be detected.

- At a center feature line the corresponding isoline disappears, thus the contribution number has to change by one crossing a center critical curve.
- At a saddle feature line the isoline splits from one to two isolines, thus the contribution number also has to change by one crossing a saddle critical curve.
- At a BSC the isolines start to split, thus the contribution number changes crossing a boundary switch boundary curve.

Further, it follows that the contribution numbers for case $n = 3 \mid m = 2$ have to be a piecewise constant function and change the value only at the mentioned curves.

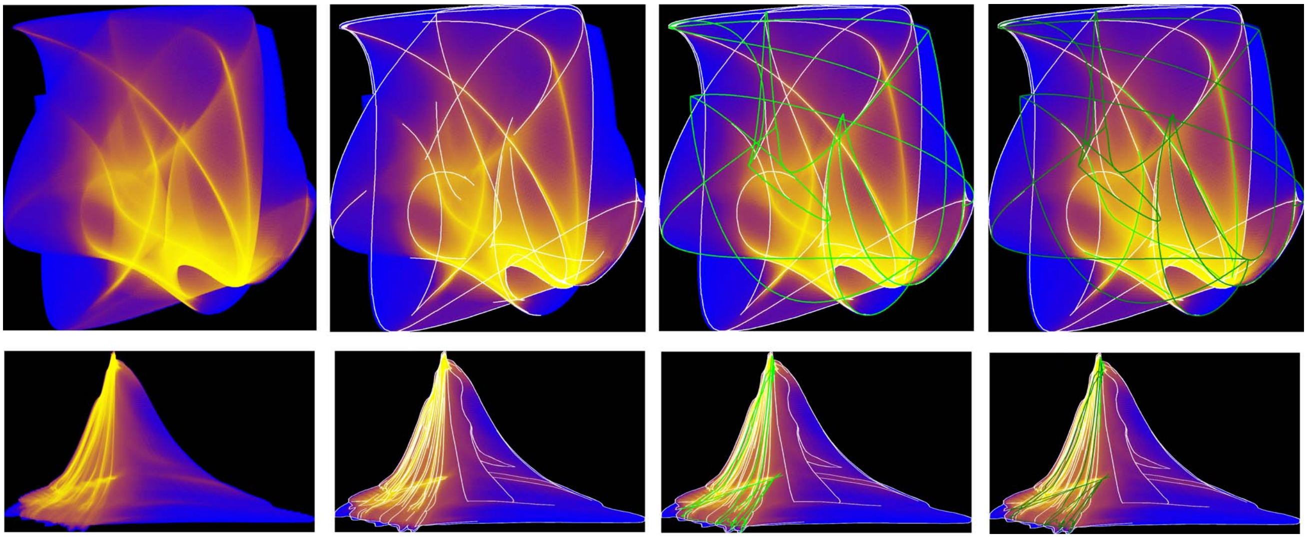

Fig. 17. Discontinuity application synthetic (up) and “Hurricane Isabel” (down) data set: (left-right) CSP, CSP with boundary switch boundary curves (BSBC) (white), CSP with BSBC (white) and box edges boundary curves (green), CSP with BSBC (white) and box edges boundary curves (dark green) and Critical Curves (green).

4.3.3 Relations between Critical/Boundary Curves and CSPs

Figure 18 shows a comparison between edges of a CSP (a) and boundary and critical curves (b). It can be noticed that edges/discontinuities of the image space of that CSP relate to critical as well as boundary curves of the case $n = 3 \mid m = 2$.

That is why boundary and critical curves are structures where the density has discontinuities. On the other hand, it has already been shown that boundary and critical curves also correspond to constant contribution number boundaries. Consequently, contribution number boundaries also relate to edges of the image space of that CSP.

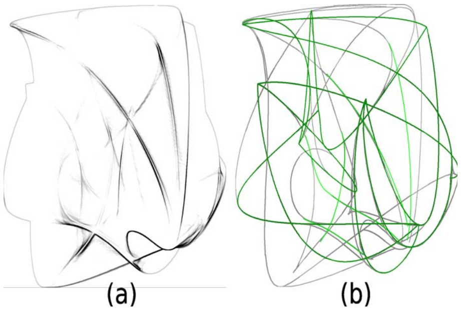

Fig. 18. Relation between CSPs and critical and boundary curves: (a) Edges in a CSP detected with a canny [5] image process operation. (b) Critical and boundary curves of the same CSP detected with the described approaches.

4.4 Application

In this section we apply the approach of discontinuities to a synthetic data set and to the “Hurricane Isabel” data set. The synthetic data set contains random scalar fields over a $30 \times 30 \times 30$ spatial domain, which is re-sampled with a uniform grid cell spacing of 0.0135. The computational time for the critical curves was 8 minutes and for the boundary switch boundary curves it was 18 minutes. The “Hurricane Isabel” data set has already been introduced in Section 3.4.3. We used the variables pressure and temperature of time step 10 with a grid resolution of $500 \times 500 \times 95$. The computational time for the critical curves was 5 minutes and for the boundary switch boundary curves it was 22 minutes.

Figure 17 illustrates the resulting CSPs and critical and boundary curves for both data sets. The CSPs have been built with an extension of the idea in [3] for the 3D case. The interpolation has been done with RBFs of Gaussian type. For the integration of the SFFF we used a fourth order Runge-Kutta scheme and a strong attractor $\alpha = 80$ value. Additionally, we used 100 as the depth of the recursion to detect seedpoints for the critical curves and the boundary switch boundary curve.

Note that these computation times can be improved by modify the attract factor α and the depth of recursion; but with more than 80% of the computational time the evaluation of the RBFs is the bottleneck of that approach. On the other hand, the computation times rather depend on the number of discontinuities in the data set instead of the number of records. The more discontinuities the more seedpoints will be detected and the larger the computational time gets.

5 Conclusion And Future Work

In this paper we introduced a discontinuity concept for continuous scatterplots of the relevant cases of $n = m = 2$ and $n = 3 \mid m = 2$. Additionally, we presented the contribution map for the case $n = m = 2$: a new visualization method for relations between the spatial domain and the data domain, and we discussed useful linking & brushing tools. Furthermore, we described detection algorithms for the mentioned discontinuities and applied them to real data sets to demonstrate the discussed approaches.

Our results show that discontinuities of the mapping function τ relate to discontinuities (edges) of the CSPs themselves and they bound areas of constant contribution number (i.e. number of connected components). This offers a deeper understanding of the underlying data sets and improves the interpretation of CSPs.

The future step will be an extension of the discontinuity concept for the time-dependent case and for continuous parallel coordinates [9].

Acknowledgments

The “Hurricane Isabel” data set was produced by the Weather Research and Forecast (WRF) model, courtesy of NCAR and the U.S. National Science Foundation, and the “north sea” data set was provided by the BSH (Bundesamt fuer Schifffahrt und Hydrographie Germany). The authors would like to thank Christian Rössl and Alexander Kuhn for beneficial discussions and Stefanie Quade for proof-reading. This work was supported by a grant from the German Science Foundation (DFG), project DFG TH692/6-1.

REFERENCES

[1] S. Bachthaler, S. Frey, and D. Weiskopf. Poster: CUDA-Accelerated Continuous 2D Scatterplots. *IEEE Visualization Conference 2009*, 2009.

[2] S. Bachthaler and D. Weiskopf. Continuous scatterplots. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1428–1435, 2008.

[3] S. Bachthaler and D. Weiskopf. Efficient and Adaptive Rendering of 2-D Continuous Scatterplots. *Comput. Graph. Forum 28(3): 743-750 (2009)*, 2009.

[4] R. Becker and W. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.

[5] J. F. Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence*, 29:679–698, 1986.

[6] H. Carr, B. Duffy, and D. Brian. On histograms and isosurface statistics. *IEEE Transactions on Visualization and Computer Graphics*, 12:1259–1266, 2006.

[7] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3d objects with radial basis functions. *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 67–76, 2001.

[8] H. Edelsbrunner, D. Morozov, and A. K. Patel. The stability of the apparent contour of an orientable 2-manifold. *Workshop Top. Methods in Data Anal. Visual. (to appear)*, pages 183–192, 2009.

[9] J. Heinrich and D. Weiskopf. Continuous Parallel Coordinates. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization / Information Visualization 2009)*, 15(6), 2009.

[10] R. Peikert and M. Roth. The Parallel Vectors Operator - A Vector Field Visualization Primitive. In *Proceedings of the 10th IEEE Visualization Conference (VIS '99)*, pages 263–270, Washington, DC, USA, 1999. IEEE Computer Society.

[11] D. Shepard. A two-dimensional interpolation function for irregularly-spaced data. *ACM '68: Proceedings of the 1968 23rd ACM national conference*, pages 517–524, 1968.

[12] H. Theisel, J. Sahner, T. Weinkauf, and H.-P. Hege H.-C. Seidel. Extraction of parallel vector surfaces in 3d time-dependent vector fields and appliction to vortex core line tracking. *Proc. IEEE Visualization*, pages 631–638, 2005.

[13] H. Theisel and H.-P. Seidel. Feature flow fields. *Data Visualzation 2003*, pages 141–148, 2003.

[14] B. Tversky, J. B. Morrison, and M. Betrancourt. Animation: Can it facilitate? *International Journal of Human Computer Studies*, pages 747–262, 2002.

[15] T. Weinkauf, H. Theisel, A. V. Gelder, and A. Pang. Stable feature flow fields. *IEEE Transactions on Visualization and Computer Graphics*, 2010.

[16] T. Weinkauf, H. Theisel, H.-C. Hege, and H.-P. Seidel. Boundary switch connectors for topological visualization of complex 3d vector fields. *VisSym 2004*, pages 183–192, 2004.