Colliding Galaxies, Rotating Neutron Stars and Merging Black Holes - Visualising High Dimensional Data Sets on Arbitrary Meshes

Werner Benger

Center for Computation & Technology
at Louisiana State University
Baton Rouge, Louisiana, USA

E-mail: werner@cct.lsu.edu

Abstract. Visualisation of datasets stemming from diverse sources is challenged by the large variety of substantial differences in topology, geometry and nature of the associated data fields. Since there is no standard on how to formulate and treat data for scientific visualisation, algorithms are frequently implemented in a highly domain-specify way.

Here we explore the potential of point-wise rendering as a generic way to represent single or multiple fields instantaneously on arbitrary mesh types. This approach is discussed within the terminology of fiber bundles as a general mathematical concept to model scalar-, vector- and tensorfields given on topological spaces (with manifolds as a particular case).

We give application examples based on data sets originating from astrophysics and show first results of a tensor field visualisation of a recently produced complex dataset of colliding black holes in their final orbit. We finally propose a data layout representing the mathematical the concept of a "field" generic enough to handle all involved cases.

AMS classification scheme numbers: 68N99, 68W99, 57-04

1. Introduction

Astrophysical data come in a large variety and tend to be huge. While many numerical simulation algorithms have been developed developed for a highly specific purpose, they intrinsically share common properties among each other, and with other scientific domains beyond astrophysics. Attempts to conceive a common data model to unify the variety of data are rare. However they become increasingly mandatory with the complexity of simulations, their resulting data, and the need of interoperability between simulation code as well as eventually with visualisation environments. (Butler & Pendley 1989, Butler & Bryson 1992) proposed to consider data sets as fiber bundles. This model was in particularly followed within the ASCI project (Cook & Matarazzo 1998). The IBM Data Explorer (Treinish 1997) (now OpenDX) is a widely known implementation of this mathematical model as applied to scientific visualisation.

Recent modern approaches build on *generic programming* to decouple algorithms from data structures (Berti 2000, Veldhuizen 1995). However, generic programming is not suitable for interactive applications where decisions about data types have to be made at runtime.

In this article, we discuss the potential of using only the per-vertex information of a mesh as a a generic approach covering many data types. Utilizing Gaussian footprints for volume rendering has been widely used for unstructured meshes (Westover 1990, Crawfis & Max 1993*b*, Crawfis & Max 1993*a*, Kreylos et al. 2000). Structured meshes are covered by texture-based volume rendering in many modern software packages (Stalling et al. 2005, Treinish 1997, Kitware 2005, Kaufman & Mueller 2004). The handling of adaptive mesh refinement (AMR) grids (constructed from a hierarchy of regular meshes) requires special concentration (Weber et al. 2001, Kähler 2005). However, these techniques are mostly suitable for rendering single scalar fields. Here we discuss visualising multiple fields (such as many scalar fields, or a multi-component tensor field). In particular, the technique to visualise a tensor field via Gabor patches, as described in (Benger & Hege 2004), is very similar to volume rendering via Gaussian footprints and therefore applicable to arbitrary meshes.

The visualisation routines described here are identical for all data types, from particle data to adaptive meshes to multiblock data. The high re-usablitity of visualisation code allows to utilise features developed specifically for a certain data type also to be applied to others, possibly leading to new insight. We demonstrate such experimental visualisation here, and intend to open new views on how to look at certain data.

1.1. Outline

Section 2 provides a basic introduction to the terms and techniques, and exemplifies visualising multiple fields based on simple artificial datasets. Section 3 demonstrates the application of these basic concepts upon recently generated astrophysical datasets: a particle system modelling colliding galaxies, a multipatch data set describing a stationary star and an adaptive mesh refinement dataset of colliding black holes. Where appropriate, the instantaneous visualisation of multiple fields is demonstrated. Finally, section 4 presents the data organisation scheme that was used to treat the diverse data fields in a common way.

2. The Technique

We demonstrate our approach to render a large variety of multidimensional data by means of a very simple artificial data set. At first, we need to specify what we mean by "multidimensional". In order to do so, we employ the mathematical concept of a fiber bundle, as it can be found in topology. In simple words, a fiber bundle is a space which looks locally like a product space. I.e., small regions of a total space E look

like small regions in the product space $B \times F$. Here B is called the base space and F is called the fiber space. The global topological structure of the total space may be different from $B \times F$. If so, it is called a non-trivial fiber bundle. The data sets that occur in scientific visualisation can well be described as trivial fiber bundles (Butler & Pendley 1989, Butler & Bryson 1992).

2.1. Nomenclature

With respect to dimensionality, we henceforth distinguish between the dimensionality of the base space $dims(B)$ and the dimensionality of the fiber space $dims(F)$. Data given on a plane or surface will thus be said to be of base space dimensionality two, data given in a volumetric domain to be of base space dimensionality three, and dynamic volumes of dimensionality four. Higher dimensional base spaces may refer to e.g. parameter studies of simulation sequences, that output evolving volumes of data, each containing a different sequence depending on some initial parameter.

At each point within the base space we will have some given data and these define the fiber space. A pure geometry (such as a mesh) will not have such fiber space data associated. If a scalar field is given on each point, the fiber space will be of dimensionality one, multiple scalar fields, vector or tensor fields will increase the dimensionality. At this point, the properties of the fiber space are not of relevance; it will in certain cases have vector space properties and is then called to be part of a vector bundle.

This scheme in its narrow sense only refers to data given on the vertices of a discretised topological space. In general, we might also have data given on different entities, such as the triangles of a surface or the cells of a mesh. Within the scope of this article, we limit ourselves to vertex-based data. In general, those can be constructed from other data, even though it might not be trivial.

2.2. Exemplification

To introduce the ideas of a systematic approach on how to deal with high dimensional data spaces, we will utilise a very simple artificial data set. It is a mere set of points arranged along the edges of a triangle - see figure 1, with three scalar fields that are given by the barycentric coordinates (r, s, t) of each point. These three coordinates fulfil the condition $r + s + t = 1$ (for a normalised triangle) and are thus interdependent.

Mapping the values of a scalar field to colours is a common approach and may yield an already three-dimensional or four-dimensional representation, when the red,green,blue channels are used independently and eventually transparency is taken into account as well. While such a three-dimensional colour mapping is an option, it is not necessarily the perceptually best one, because not all colours are perceived equally by the human spectator. Thus, linear data may appear highly non-linear, and may lead to visual artefacts. In the following we will use only one-dimensional colour mappings, i.e. all colours will refer to the same one-dimensional data field and only represent different values within.

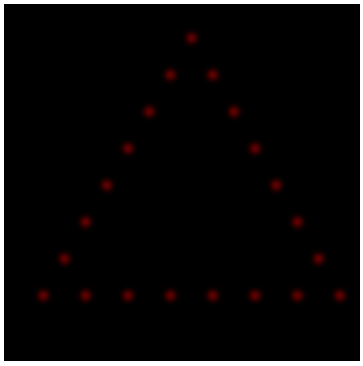

Figure 1. Simple point set that will be used for illustrations. At each point, there are multiple data sets defined, and we seek for methods to display them instantaneously.

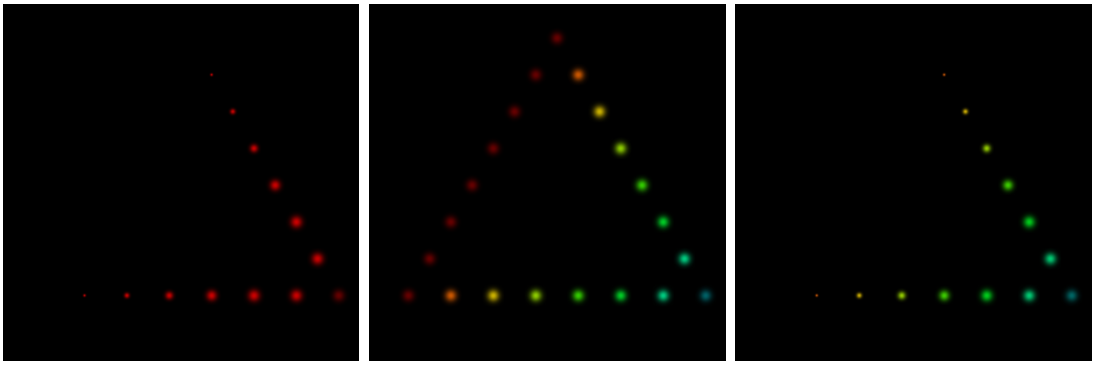

Figure 2. Point set with scalar data, displayed as variation of point size or colouring; both methods combined yield a visualisation of two scalar fields.

Beside colouring, when representing singular points we have the freedom on how to display each. Instead of a one-pixel representation that is hardly visible, we use a Gaussian intensity distribution as a natural choice here. Such can be rendered by modern graphics hardware extremely well using so-called point sprites. The half width of the Gaussian sprite is now a free parameter that can be used in addition to the colouring, and be used to represent a two-dimensional fiber space, as in figure 2 (complementary to vector arrow icons, who as well represent a two-dimensional fiber space, but are more suitable for vector spaces than general fiber spaces).

Further degrees of freedom are available via the geometric location of the vertices itself. When representing the vertices of an n-dimensional base space within in an m-dimensional embedding space with we have m degrees of freedom to represent data on the vertices by shifting their coordinate locations. In general, this will however lead to ambiguities if not the original coordinate locations can be clearly distinguished from the shifted ones. If the embedding space is higher-dimensional than the data base space, using the additional dimensions is always unique and can be used without ambiguity concerns (for instance, a height field of a two-dimensional data set in a 3D graphical representation). Example given by figure 3 where we also have a unique direction of the additional dimension within 3D space, as given by the plane of the triangle that defines

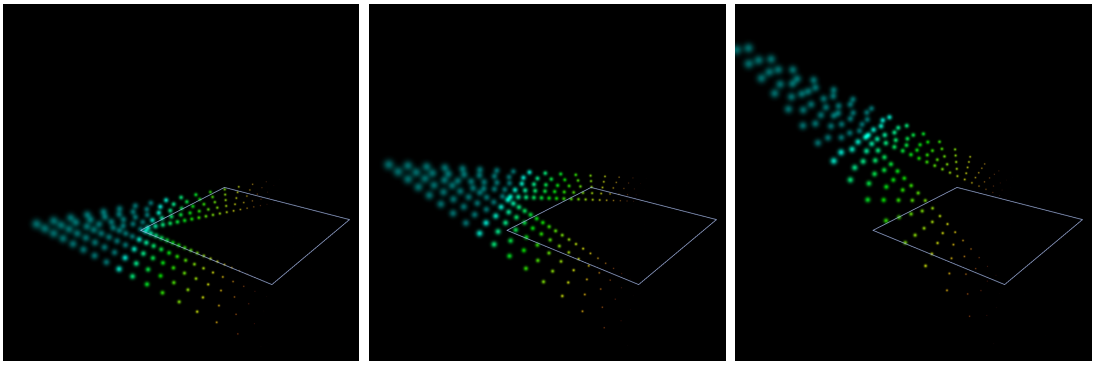

Figure 3. Using geometrical variance to display scalar data. The same field is redundantly mapped to colour, point size and elevation (with increasing scaling parameter of the elevation for purpose of illustration).

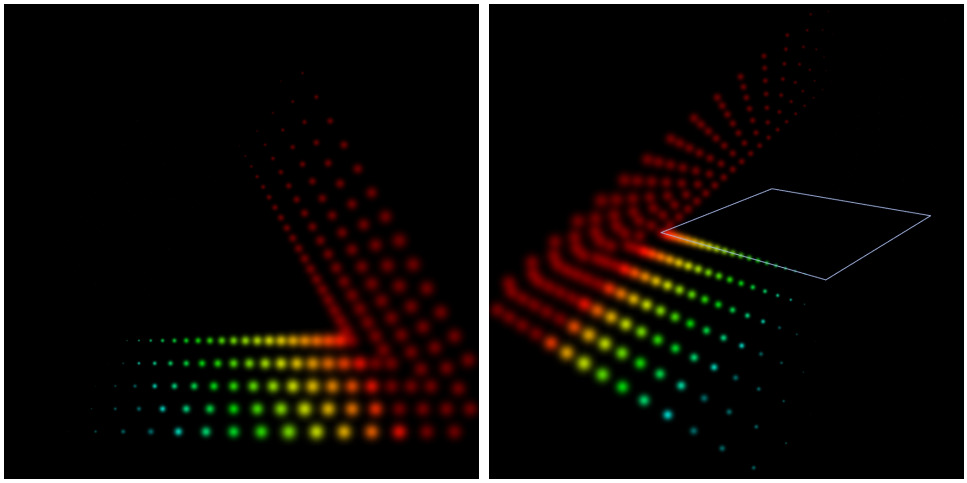

Figure 4. Using point size, colouring (left) and geometrical variance (right) to display three independent scalar fields on a set of point, thus eventually yielding a "visualisation of a three-dimensional fiber space over a two-dimensional base space".

the two-dimensional base space.

Finally, figure 4 displays all three scalar quantities as given on the triangular test case by independent means of size, colour, and "elevation" as form of varying the geometric location of the point locations. While in this case the concept of "elevation" is uniquely defined, it will however be demonstrated in the text below that the ambiguity of varying a geometric location within the coordinate space of the vertices itself is not as bad as one might believe initially.

2.3. High-Dimensional Fiber Space

Varying the size of a spherical Gaussian distribution function represents a one-dimensional fiber space. More complex shape functions may be utilised to represent higher dimensional fiber spaces. For instance, also the half-width can be varied, or an additional radial frequency imposed, such as with an Airy disk (the diffraction pattern

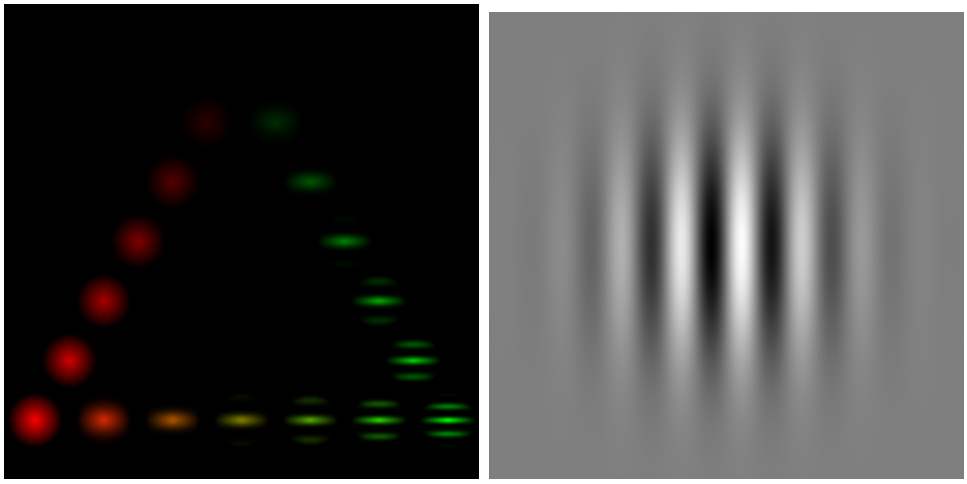

Figure 5. Visualisation of a five-dimensional fiber space (a symmetric tensor field of second order, the technique of "tensor patterns" - left figure) via parameterized Gabor patches (right).

cast by a point light source). Non-symmetrical modifications of the point shape are ideally used to represent an intrinsically geometric property of the data field in the fiber space. For instance the direction of a vector field at each point is well represented by a vector arrow icon or vector splat (Crawfis & Max 1993*b*)), the metric properties of a tensor field of second order is well represented by the cross-section of a quadric surface.

One such specific technique was presented in (Benger & Hege 2004) and more thoroughly described in (Benger et al. 2006) by means of medical data sets. It was substantially inspired by (Laidlaw et al. 1998). This technique called "Tensor Patterns" allows to represent six independent, but related quantities (such as the components of a symmetric 3×3 tensor field) at each point in space. Instead of a one-parametric radial Gaussian distribution it employs Gabor patches, which are used as an optical stimulus in vision theory - see figure 5, right. This visual primitive allows to encode more parameters per point than a pure Gaussian blob, as demonstrated in figure 5 (left).

An important benefit of point shapes with smoothly increased transparency at their boundaries, such as Gaussian blobs and Gabor patches, is their property of smoothly blending when scaled large. The result is a visually pleasing smooth impression of a continuous dataset as in figure 6. While information about a specific data point is less comprehensible by such overscaling, the overall global structure comes out more clearly. Such methods are thus suitable for "data mining" purposes and first inspection of yet unknown datasets in order to identify global features and structures.

2.4. High-Dimensional Base Space

A natural approach to deal with high dimensional base spaces is to slice them into lower-dimensional domains and do analysis of the resulting fiber space on this subdomain (figure 7, left). While depicting each data point as geometric point is not helpful per

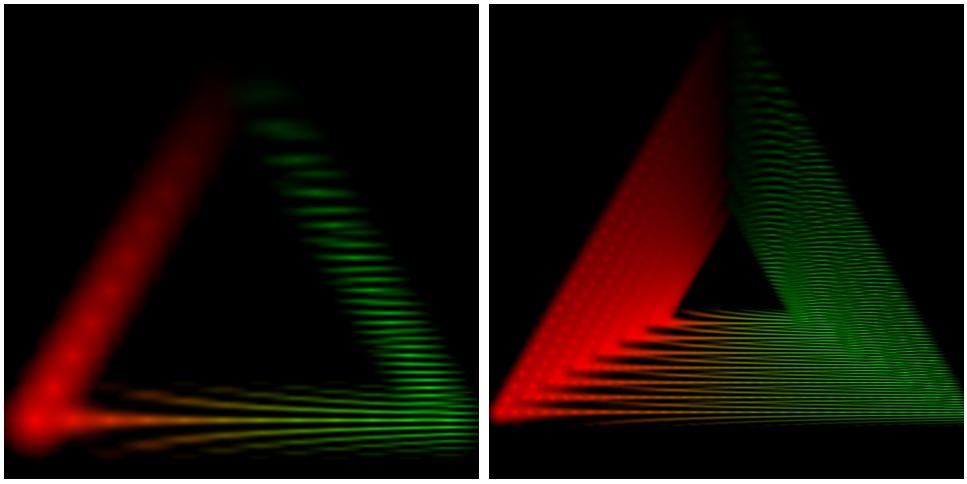

Figure 6. Scaling of the vertex primitives allows a nearly continuous impression of the data set (left: same number of points as in figure 5, right: increasing number of points), suitable as visualisation of a five-dimensional data set, e.g. for data mining purpose and reading off global structure variations.

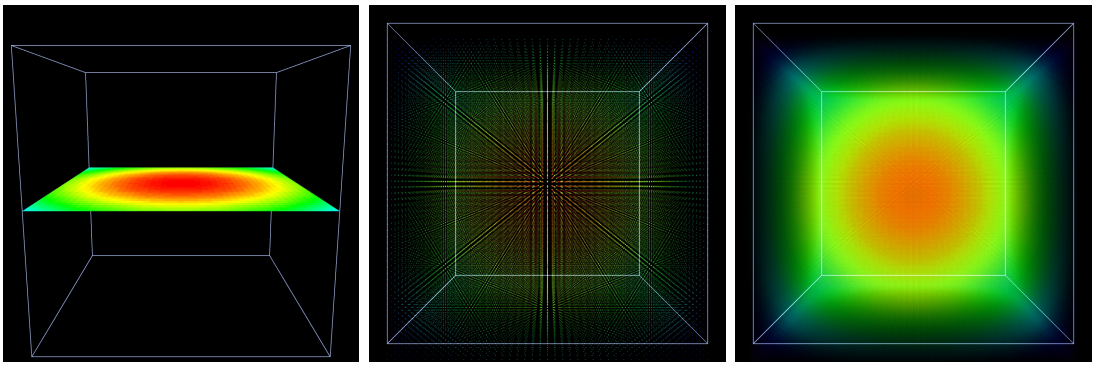

Figure 7. Inspecting a three-dimensional base space: via textured slice, as single points, as Gaussian point sprites scaled to blend with their neighbours (also known as volume rendering via splatting using Gaussian footprints).

se (figure 7, centre), the ability to employ smooth scaling of the primitives yields an effect known as volume rendering via splatting using Gaussian footprints (Kaufman & Mueller 2004). While volume rendering is a 1:1 representation of a entire scalar-valued data set, its applicability highly depends on the complexity of the structures found in the data sets. Finding an appropriate colour mapping scheme (also known as transfer function) to bring out these features best is a research topic by itself (Kaufman & Mueller 2004, Kindlmann & Durkin 1998).

As a technique that merely uses the geometrical location of the data points, this variant of volume rendering can directly operate on an arbitrarily discretised base space. In contrast, texture-based approaches of volume rendering require a regular grid structure of the base space, by the benefit of allowing interpolation between the data values to achieve visually higher quality. Independent from the distribution of vertices in coordinates, the data points to be displayed can be easily restricted to a subdomain

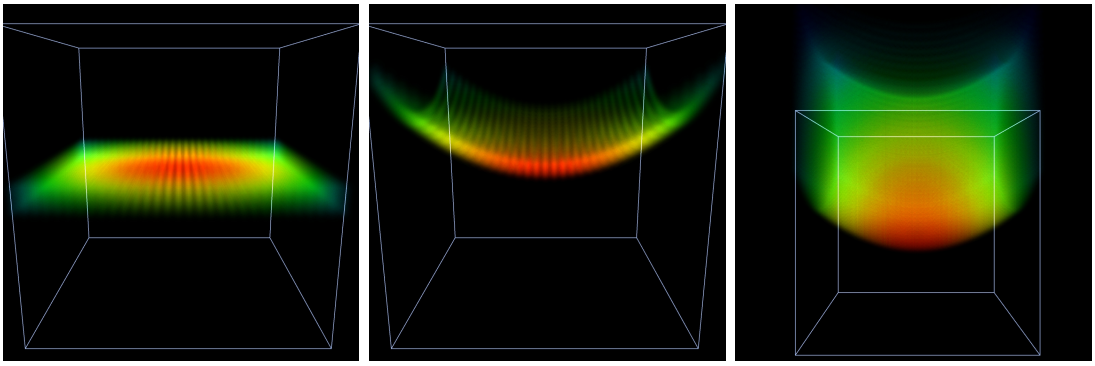

Figure 8. Reduction of three-dimensional base space by arbitrary geometric cutting, adding geometrical variance, exploring full volumetric geometrical variance.

via by the graphics hardware, thus achieving an effect of a planar slice but on arbitrary complex base space discretisation schemes (more illustrative examples to be given in the next section). By defining such a subset of the base space, geometric deviation of the point location can then be used in addition to pure colour mapping, yielding the effect of a height field or elevation map. To a certain extent, this approach even works when applied to the entire volume, as illustrated in figure 8.

3. Applications

In the previous section, the notions of base space and fiber space with their respective dimensionality have been introduced, and a systematic approach on how to visualise them via point-wise rendering. This approach is extremely fast, as data can be more or less directly transferred from the hard disk to the graphics card. Pre-processing is thus minimal, and interactive browsing of a high-dimensional parameter space - such as an evolution sequence over time - is achievable when utilizing appropriate caching. While the data sets in the introductory section have been trivial, we will now show the performance of the discussed techniques by means of three intrinsically different data sets originating from practical astrophysical research. It is to be noted that the rendering code for each of these diverse datasets is identical.

3.1. Particle Systems - Colliding Galaxies

The first data set explored here models two galaxies on collision course. It has been described by (Kapferer et al. 2005). The entire simulation covers about 10^9 years and required one week of computational time on a 40-Core Opteron cluster. The simulation code is based on Gadget 2 (Springel 2005) and implements an N-body gravitational model coupled with smoothed particle hydrodynamics (SPH).

The data set consists of 1.6 million points without explicit neighbourhood information, distributed over 100 time steps. Several scalar fields are given on each point, these are: mass `mass`, internal energy `U`, density `rho`, gravitational potential

`pot`, SPH smoothing length `hsml`, resulting in 17GB when stored in HDF5 (The HDF5 Group 2008). The simulation distinguishes among four kinds of particle types, describing gas, dark matter in the halo of the galaxy, existing stars in the disk of the galaxy and newly generated stars that arise from overdensities in the gas. In the fiber bundle terminology, this is a four-dimensional base space with a five-dimensional fiber space.

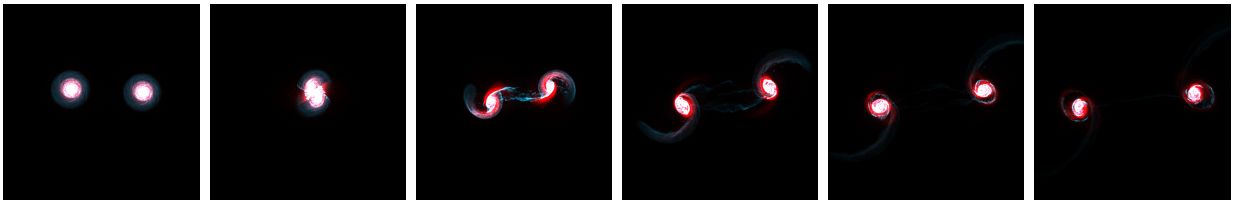

Figure 9. Evolution of colliding galaxy dataset, smoothed particle hydrodynamics (SPH) simulation based on 1617286 points, using internal energy U for colouration.

In the visualisation shown here, we do not distinguish among these particle types, and seek to display the particles merely based on their numerical scalar fields given on them (though the particle type could however be seen as an integer-valued scalar field on the particles). Figure 9 displays six timesteps from the evolution sequence, using the internal energy U for colouration. These images are considered a "first time view" of the data, without adjustment of colourmaps to the data range.

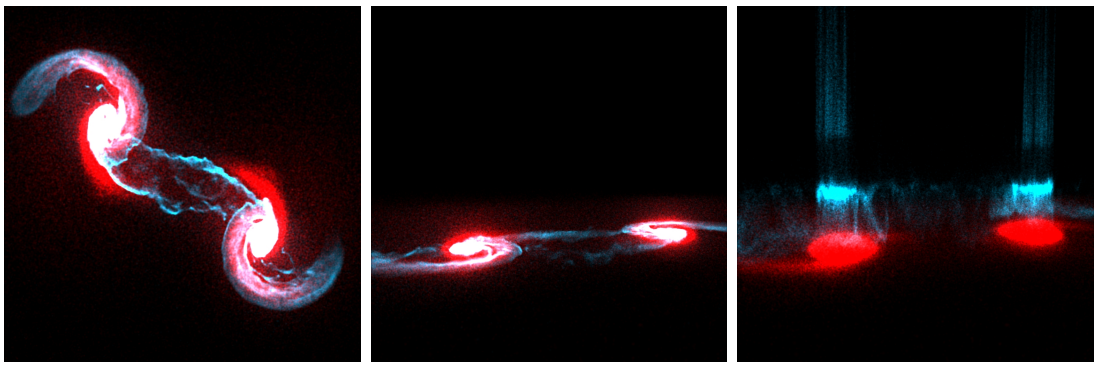

Figure 10. Nearly monochrome rendering of the three-dimensional point distribution. It is dominantly concentrated around a common plane, thus we can shift the point locations from this plane based on data values. This is similar to an elevation field on surface data, but here employed upon a particle set.

While the dataset comes with fully three-dimensional coordinate locations, the physics of this very scenario result in a distribution of the particles dominantly in the same plane, as depicted by the nearly edge-on view in figure 10. This finding allows to distribute the particles similar to an elevation field over a plane, and results in a geometrical representation of the scalar field, as in figure 10, right. While this approach is ambiguous with respect to the point locations, the overall properties of the scalar field is still easier to perceive than via a colourmap since linearity is better sustained.

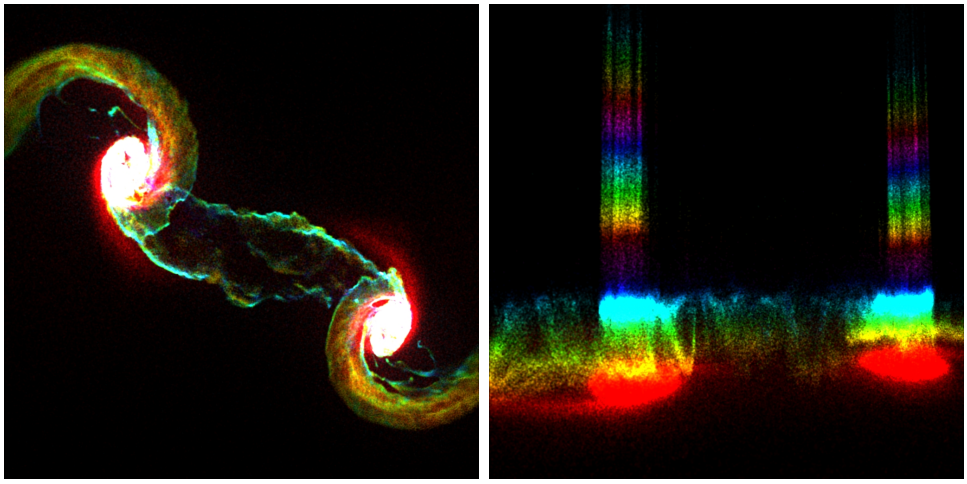

Figure 11. SPH dataset carrying internal energy U rendered with mere colour mapping (left), and additional variation of geometry (right).

A more varying colourmap as in figure 11 reveals variation of the scalar field among the point locations, yet the geometrical variation ("particle elevation") depicts the scalar field more clearly. As in Figure 11, right, both colouration and displacement redundantly refer to the same scalarfield U.

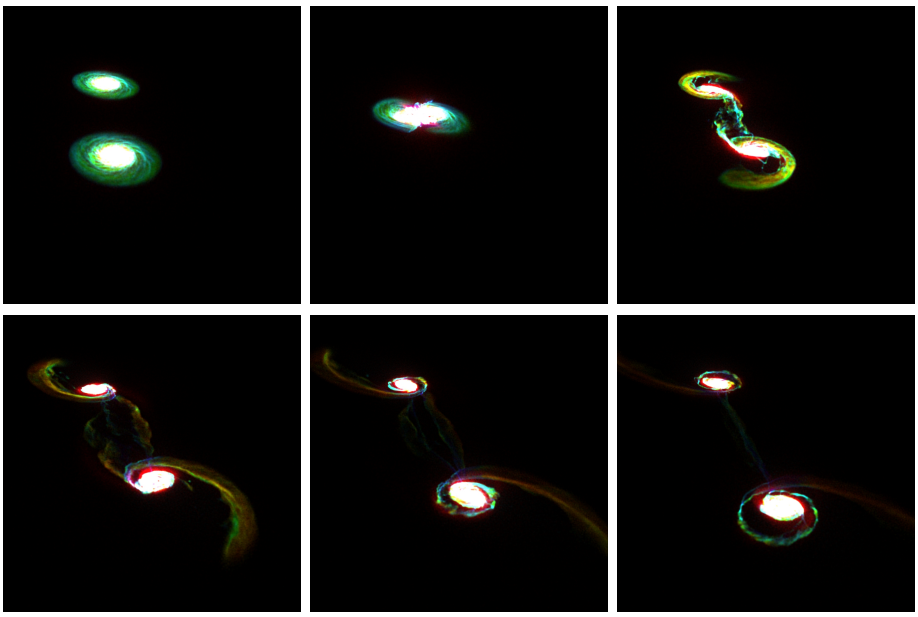

Figure 12. Coloured Evolution of SPH Colliding Galaxy Dataset

Finally, we compare an evolution sequence of the galaxy dataset, as visualised by pure colour mapping figure 12 versus adding geometrical displacement as in figure 13. While the pure colourmapping results in a more "natural" representation of the data, the elevation method might be superior to appreciate the properties of the data field. It is not within the scope of this article to provide a physical interpretation here, but

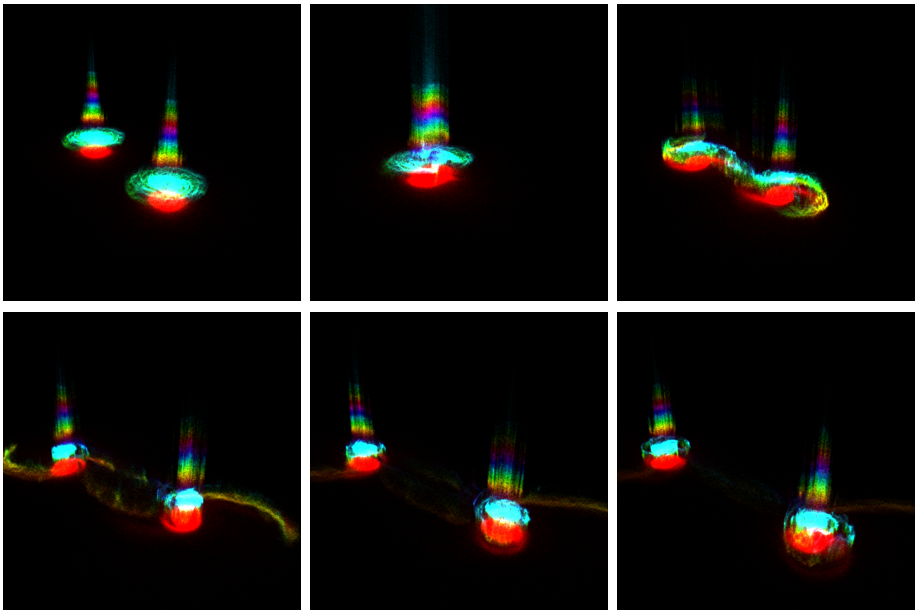

Figure 13. Animation sequence of the galaxy dataset using colour and geometry variation.

to demonstrate some ideas that may be useful for visual data analysis.

As we have two ways to represent a scalar field on a particle set, we may represent two independent fields this way. Depending on which field is mapped to which representation, the results may be easier or more difficult to grasp visually, as demonstrated in figure 14. The geometrical deviation is clearly perceptually overwhelming the colouration. Rendering the gravitational potential by elevation and internal energy U by colouration (figure 14, lower right) tells us that particles of similar gravitational potential still may have significant variations in internal energy. This relationship is not obvious from the alternative choice as in (figure 14, lower left).

We may now explore all of the six scalar fields given on the particle set using colouration and displacement redundantly, as shown in figure 15. The displacement yields a visually strong impression and provides an overview of "clusters" of similar data values superior to mere colouration, in particular, since finding perceptually equidistant colours for the colour transfer function is non-trivial, see e.g. (Teufel & Wehrhahn 2000). By introducing the point size as an additional visualisation quantity as well, we arrive at the possibility to depict three scalar fields instantaneously. The point size is perceptually weak by itself if single points are not distinguishable such as in figure 16 (left). However, the size of the splatting elements still contributes essentially to the appearance of the overall image by weighting certain regions. In the setup used in figure 16 we see particles in a strong gravitational potential, while those in weak potential are visually suppressed. Density variations in the particles are depicted in the colouration, whereby the additional geometrical displacement representing their masses shows that we have to deal with mainly two classes of massive particles here. Finally, we inspect an evolution sequence of the data using the setup to visualise the three-dimensional fiber space, as in figure 17.

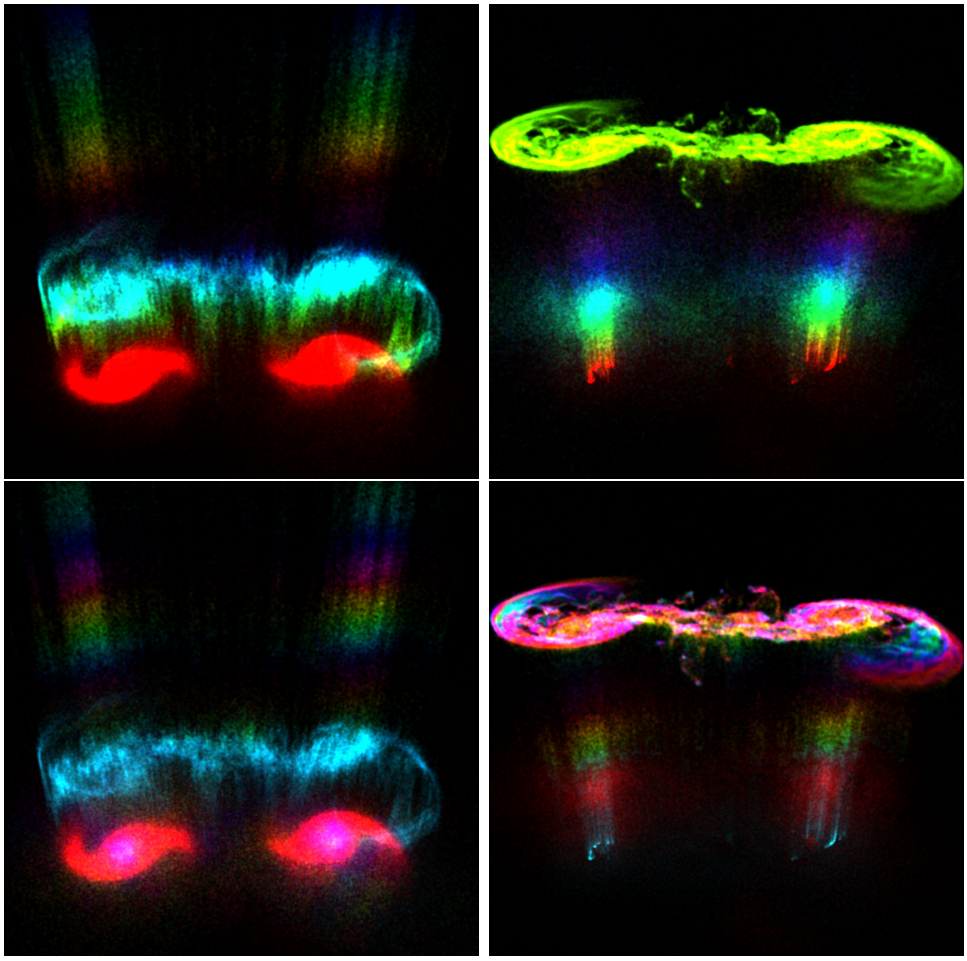

Figure 14. SPH Dataset, internal energy (U), potential energy (pot), elevation by U + colour by pot, elevation by pot + colour by U.

Without a-priori assumptions of the properties of the particles, it allows us to grasp the different densities of two classes of particles that reside in a large gravitational potential. With appropriate caching mechanisms, rendering of the entire evolution can be done interactively and in realtime. This includes variations of different mappings from scalar fields to rendering parameters.

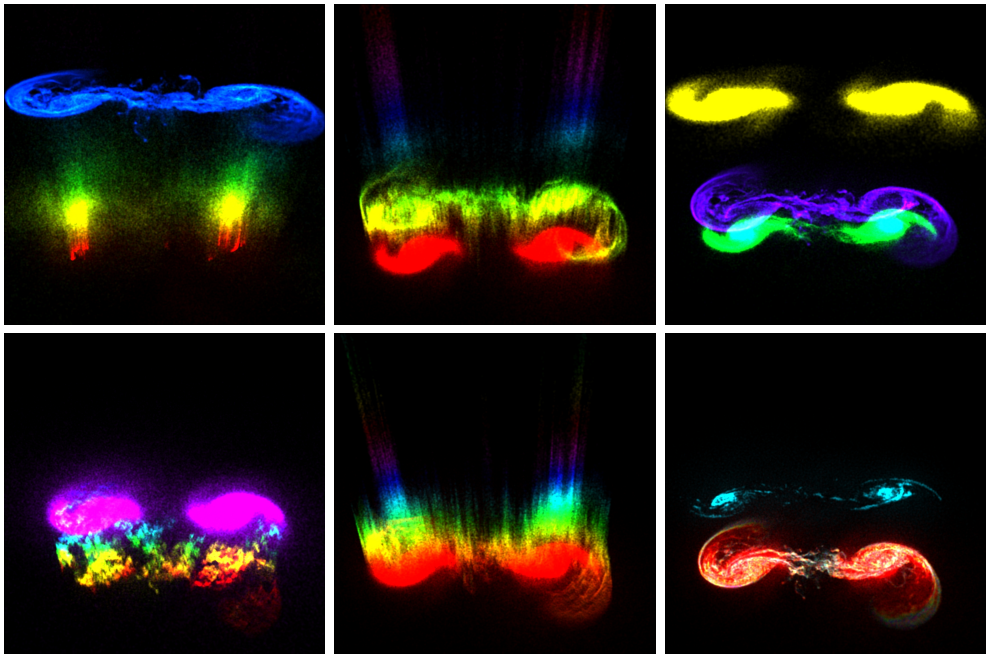

Figure 15. Display of gravitational potential (pot), internal energy, mass, ne, rho, hsml. Colouration and elevation displacement are used redundantly for each scalar field.

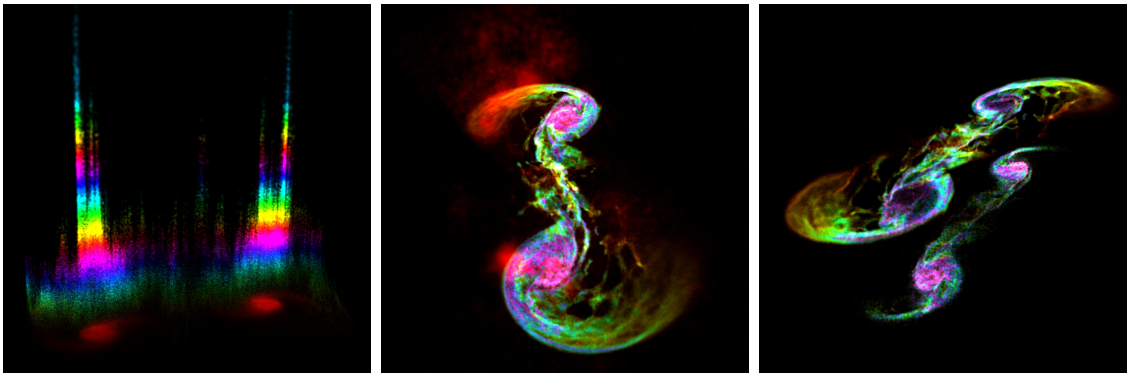

Figure 16. Exploring the three-dimensional fiberspace of a three-dimensional particle set: single field (rho), set triply redundant as point size, geometry, and colour; two scalar fields using point size(pot) and colour(rho); three scalar fields using size(pot), colour(rho) and geometry(mass).

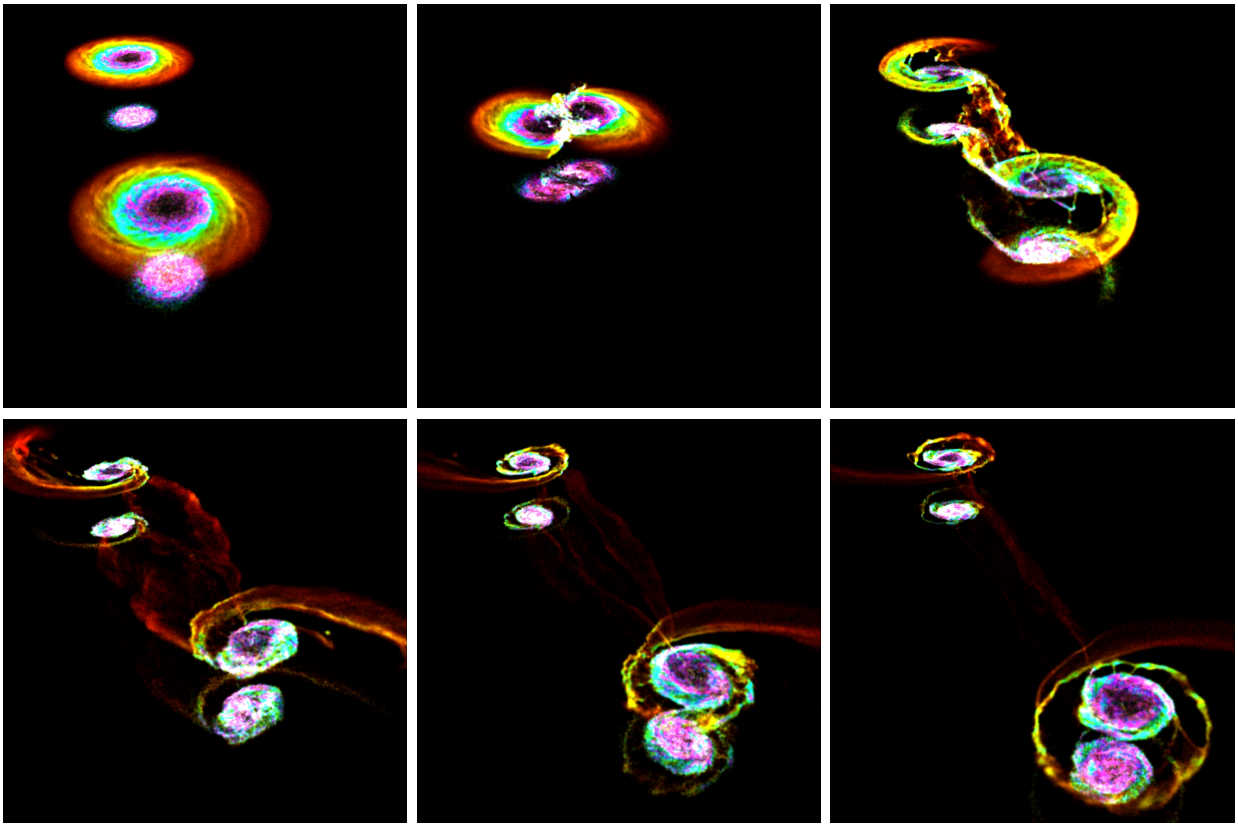

Figure 17. Animation sequence of Galaxy Dataset including colour(rho), size(pot) and geometry variation(mass).

3.2. Multipatch Data - Rotating Neutron Star

Many systems of interest in computational fluid dynamics as well as general relativistic astrophysics and other domains benefit from using non-cartesian coordinates for computational grids. Such allows continued use of topological simple discretisation schemes, i.e. regular (structured) meshes in the form of multidimensional arrays, thereby avoiding the complex topology of e.g. triangular or tetrahedral meshes.

It is however not the best choice to cover an entire domain with the same coordinate system, due to the need to avoid coordinate singularities and degenerated grid points (multiple vertices in the grid referring to the same point in physical space). In general relativity, a global coordinate system might not even exist at all.

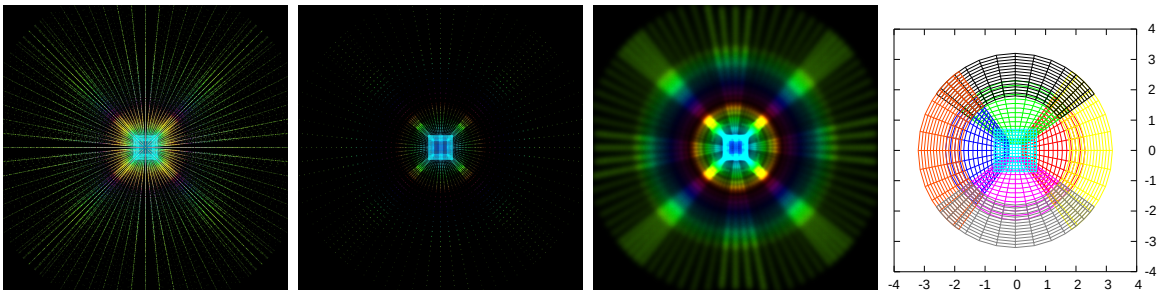

Figure 18. Multipatch volume, constructed from 13 distinct regular patches. Left image: depicting each point of the volume; centre image: reduction to an equatorial slice; centre right image: scaling of point size by radial distance; right image: patch layout by coordinates (Zink et al. 2007).

Spherical polar coordinates have singularities at the poles. A sphere cannot be covered by a single regular mesh without resulting in degenerated points. Addressing this issue requires at least two regular meshes, one for the northern and one for the southern hemisphere - with arbitrary overlap at the equator. Each of those meshes covering a coordinate domain is called a *patch*, and a dataset consisting of many of them is then called a *multipatch* dataset. While the topology of each patch is trivial, the global topology is no longer trivial, which troubles visualisation algorithms.

In a three-dimensional setup there may easily be much more than just two patches required, and here we demonstrate a visualisation based on a dataset built from 13 separate patches, as described in (Zink et al. 2007). This specific dataset describes a stationary rotating neutron star, resulting in 350 time steps of a single scalar field (density). The ca. 300.000 data points are distributed over 13 patches with 4 blocks each (due to parallelisation issues). Figure 18, left, depicts all points in the volume, which can be easily restrained to just a plane in world coordinates. In order to achieve a smooth appearance, the size of each Gaussian footprint needs to be related to the distance of a point to its neighbours; in this specific case, this distance is coarsely related to the radial coordinates. The regions of patches overlapping each other appear brighter. In the fiber bundle terminology, this is a four-dimensional base space (provided in multiple coordinate systems and patches) with a one-dimensional fiber space.

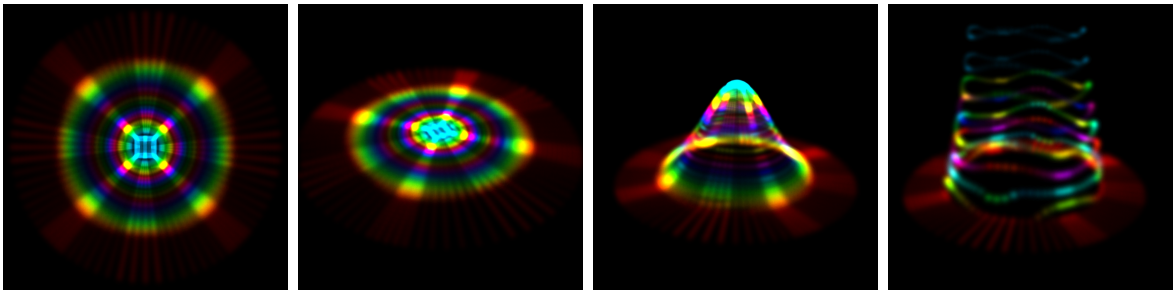

Figure 19. Rendering of the vertex plane rendered as gaussian point sprites (left), perspective rendering (centre left), geometry variation (centre right) and high sensitive colours and geometry variation (right).

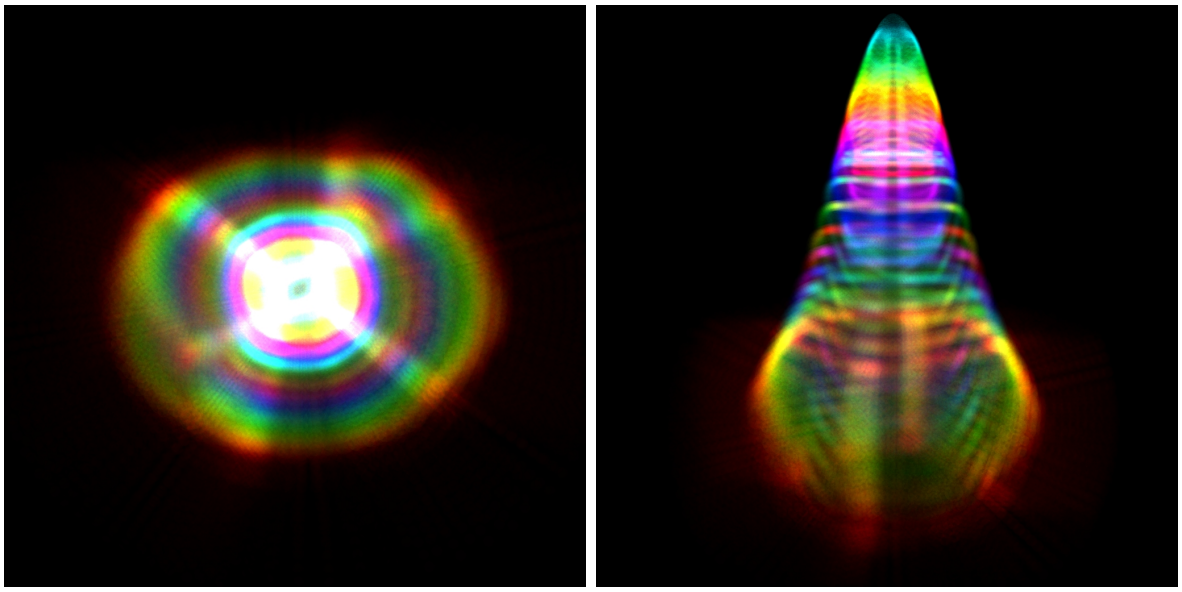

Figure 20. Rendering of the entire multipatch volume, setting points in place (left) and imposing additional geometry variation (right).

Restricting the rendering of points to a plane is efficiently done by graphics hardware and yields an impression similar to a 2D slice of the multipatch volume. However, no knowledge of the complex topology is required and this approach is very fast, thus suitable for rendering of time-dependent data (four-dimensional base space). By shifting the render points in coordinate space, the appearance of an height field is achieved, as demonstrated in figure 19. Proper scaling enhances even small variations, showing tiny instabilities in the otherwise stationary rotating neutron star. This topology-independent point-wise rendering intrinsically morphs to volume rendering, as the thickness of the rendering domain is a free parameter. Imposing a geometrical shift to each data point may even work when applied to the full volumetric volume, as depicted in figure 20. At the moment of writing this article, only one scalar field had been available in the data, so the application of a two-dimensional fiber space could not be demonstrated.

3.3. Adaptive Mesh Refinement - Colliding Black Holes

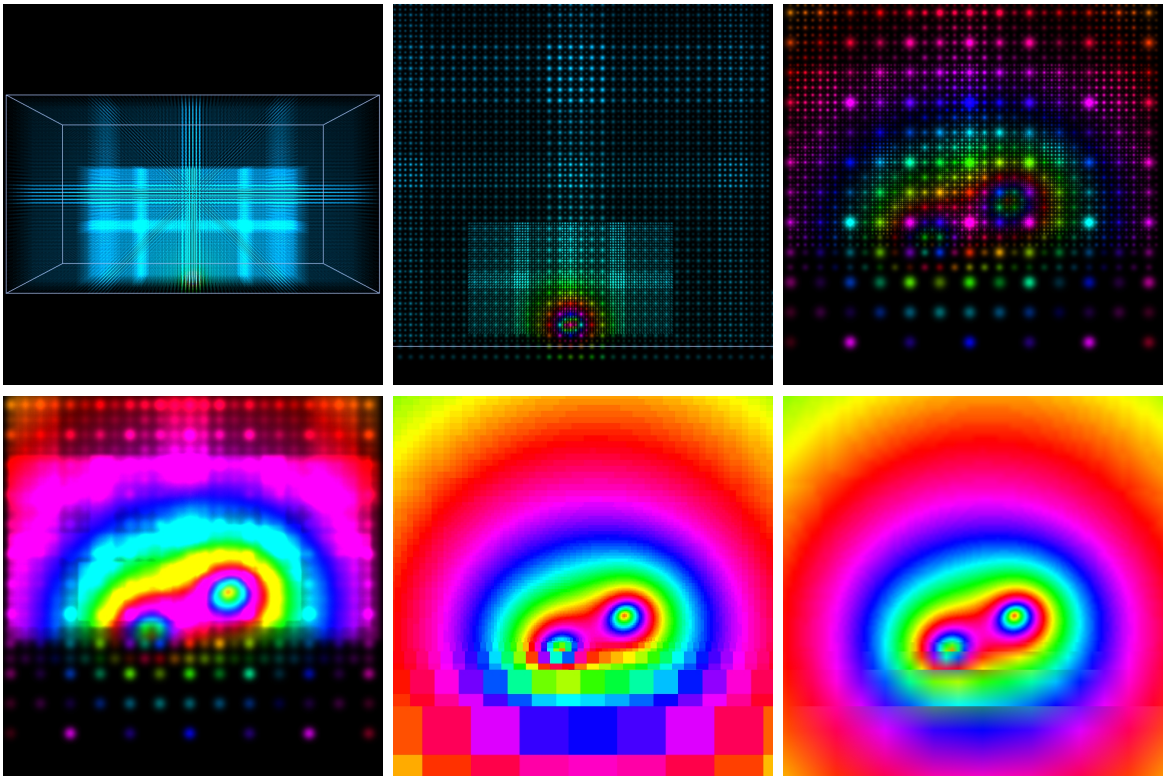

Figure 21. Rendering of a three-dimensional AMR dataset: Upper row: full volume, reduction to a 2D slice, detail depicting refinement levels; lower row: proper scaling of Gaussian splats, texture-based alternative without and with interpolation.

Orbiting and coalescing binary black hole system emit gravitational waves, which will be detected by gravitational wave observatories such as LIGO (LIGO 2008) and GEO600 (GEO600 2008). Both are coming on-line now. In the past three years, advances in numerical methods made it possible to simulate such systems numerically in a stable manner and with unprecedented accuracy (Campanelli et al. 2006, Baker et al. 2006, Diener et al. 2006). One prototypical setup of such a binary system is *QC-0*: This setup performs approximately one orbit, then merges, and finally settles down to a single rotating black hole, emitting a burst of gravitational radiation in the process. Current state-of-the-art simulations of black hole binaries also include several orbits before merger, spinning black holes, spin/orbit interaction, and recoil from asymmetric gravitational wave emission. These simulations were impossible without the Adaptive Mesh Refinement (AMR) scheme; we demonstrate visualisation of a *QC-0* data set.

AMR is a technique to automatically adapt the resolution of structured grids to a given function by recursively overlaying fine grids over coarser grid where necessary. The Berger-Oliger AMR algorithm (Berger & Oliger 1984) is a landmark achievement without which many numerical calculations would be plainly impossible. So-called *refinement criteria* dictate where fine grids should be placed, and as these criteria change

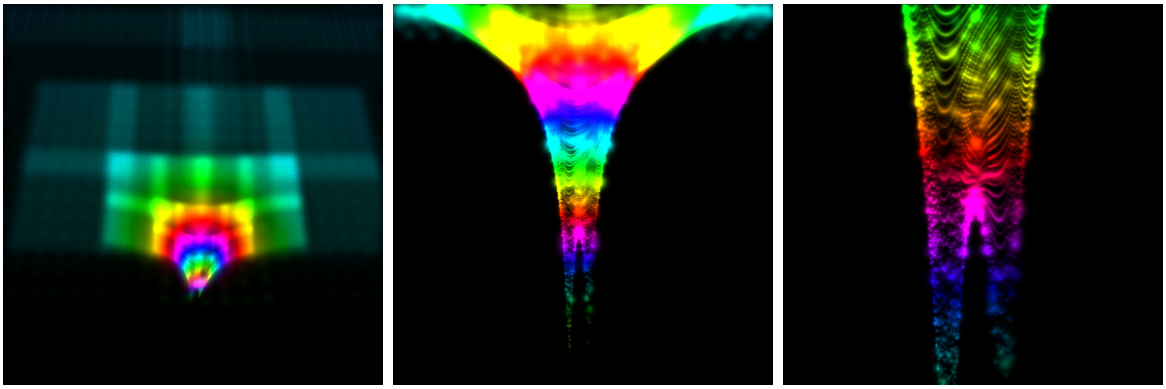

Figure 22. Geometric Elevation of vertices within of AMR volume, based on the "lapse" scalar function representing time.

during a simulation, fine grids are moved, created, or destroyed. The main advantage of AMR over unstructured grids is that structured grids can be handled much more efficiently on current hardware, and that numerical stability may be easier to achieve. AMR was introduced to three-dimensional calculations in numerical relativity e.g. in (Schnetter et al. 2004), which also gives more details on the time stepping algorithm.

Numerical stability requirements make it necessary to timestep the solution on finer grid in smaller steps than on coarser grids, leading to a rather complex, recursive time evolution algorithm where not all grids exist at all times. This in turn complicates post-processing and visualisation, which generally requires interpolation (in time) between successive coarse grid points if no fine grid points exist at a certain time. AMR was introduced to three-dimensional calculations in numerical relativity e.g. in (Schnetter et al. 2004), which also gives more details on the time stepping algorithm.

While each structured grid by itself is topologically simple and the comes with a linear mapping to coordinate space, the global structure is no longer trivial, which impedes efficient visualisation algorithms such as texture-based volume rendering (Kähler 2005, Weber et al. 2007). An AMR dataset may however always be treated as an unstructured grid and volume rendering of scalar fields may be achieved with Gaussian footprints about the size of the voxels of each refinement level. Reduction of the rendering domain to a thin plane yields results similar to texture-based rendering approaches (figure 21). With the point size being fixed in the case of rendering AMR data sets, geometric variation of point locations may still be used to enhance the properties of the data set figure 22.

The primary quantity in general relativity to consider is the metric tensor field. Here we have to deal with the a four-dimensional base space (the spacetime) and a 10-dimensional fiber space (if no matter is involved) when the full four-dimensional metric tensor is used, or a 6-dimensional fiber space when only the spatial part of the metric is investigated. However, the direct visualisation of such tensor fields is not common practice, once due to the practical availability of tensor field visualisation tools to end-users, and secondly due to the unfamiliarity of possibly end-users with the

complex visible result. Visualisation of tensor fields is mostly investigated in medical imaging based on magneto-resonance data acquisition methods (Benger et al. 2006). Such techniques tend to be highly domain-specific and do not necessarily transport over to astrophysics, but provide good source of inspiration, see (Benger 2004) for an extensive discussion. Dealing with a base space in the layout Adaptive Mesh Refinement however adds another hurdle, and early results of tensor field visualisation on AMR are presented here the first time.

It is common use in numerical relativity to decompose the four-dimensional metric tensor of the spacetime (10 independent quantities) into the lapse scalar α, the shift vector β and a spatial metric tensor γ. In figure 21 and figure 22 we displayed the lapse function as scalar field, as it represents the amount of time between two subsequent time steps of the simulation. This lapse function is an indication of the gravitational potential. Gravitational redshift can be intuitively explained as light loosing energy when escaping from a gravitational potential, but as well as time progressing slower in regions of extreme gravity. In the extreme case of a black hole, time comes to an apparent hold at the horizons of the black holes, which is hereby represented by the lapse function dropping to zero - no advancement of the time coordinate at the location of the black hole horizons in the numerical grid.

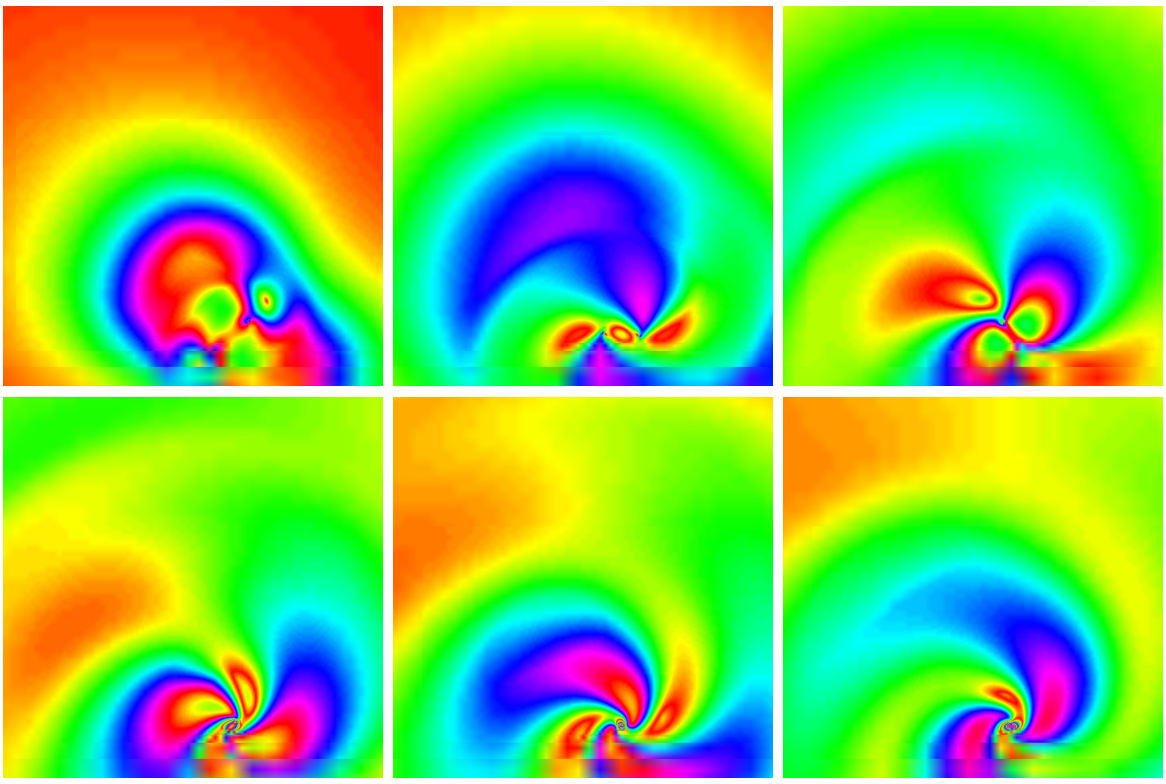

Figure 23. The linear shape factor of the metric tensor field of a black hole merger, as it depicts the creation and evolution spiral structure indicating a gravitational wave.

The full complexity of the spacetime however requires inspecting the metric tensor field. Inspecting the components of the metric tensor as scalar fields is highly misleading,

since those depend on the coordinate system, and are thus view-dependent (i.e., not necessarily of physical relevance by itself). In order to apply scalar field visualisation techniques, we need to extract certain invariant quantities. (Westin et al. 1997) introduced geometrical measures for the local properties of a tensor field, the so-called shape factors. They describe the relationship of the eigenvalues of a tensor field: the linear shape factor is large when one eigenvector is dominant, the planar shape factor indicates two dominant eigenvectors and the spherical shape factor is a measure for isotropic regions (all three eigenvectors of equal weight).

Figure 23 displays the linear shape factor - a concept originating from medical imaging - on the astrophysical dataset of a black hole merger. In contrast to visualizing the metric components or the lapse function, the creation and evolution of a spiral structure indicating a gravitational wave becomes evident immediately. The nature of the gravitational wave can be interpreted as a region in space that is predominantly "stretched" in one direction. While the linear shape factor indicates where such linear

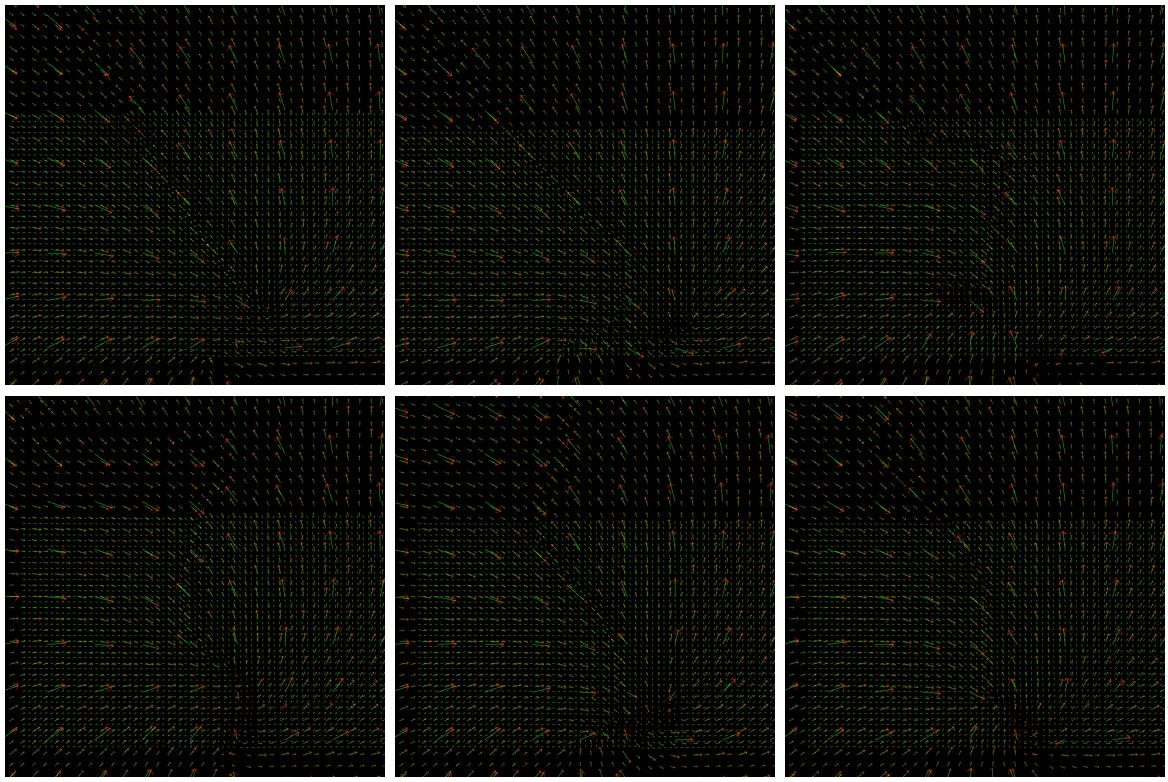

Figure 24. Eigenvectors of the metric tensor field, same time steps as in figure 23, shown here for illustrative purpose; three independent values given per point. However, rendering via vector arrow icons is not appropriate.

stretching happens, it does not depict in which direction it occurs. Such information if available from the major eigenvector of the metric tensor field, displayed in figure 24 using the standard approach of vector arrows at each point (as a visualisation of a three-dimensional fiber space). The visual results are not very impressive and shown here just for illustrative purpose. Actually of interest is to completely convey all properties of the

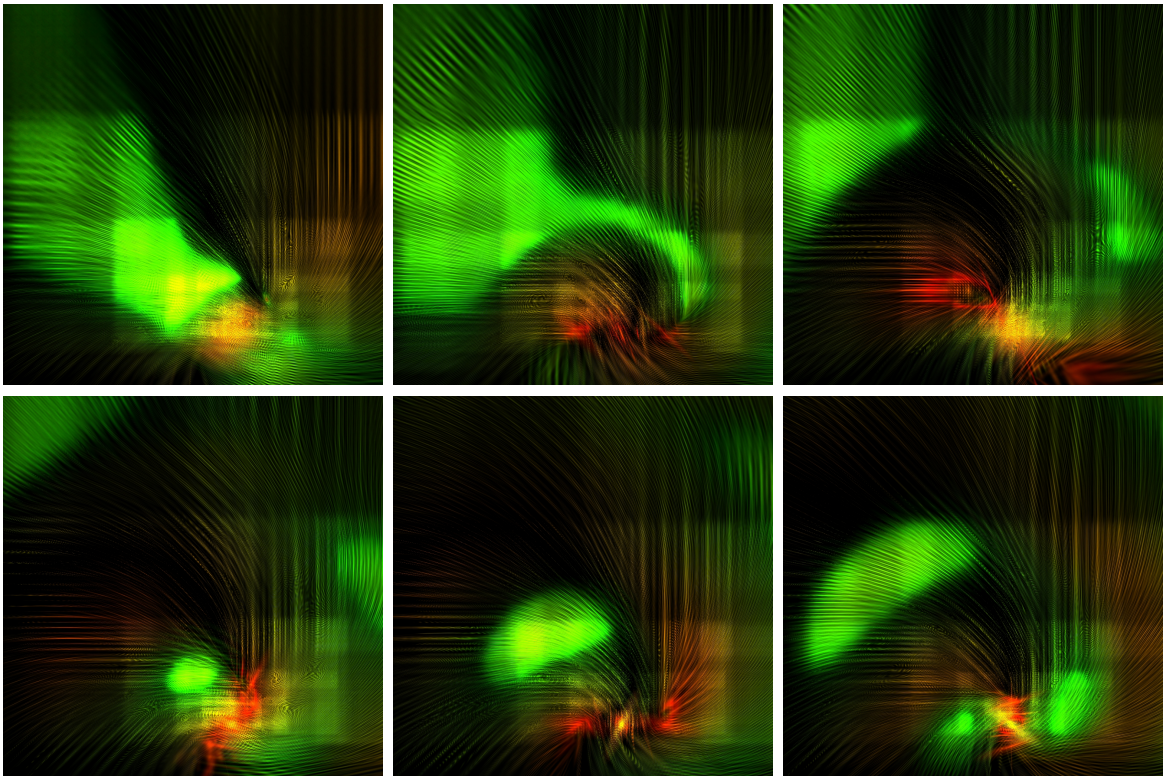

Figure 25. Metric tensor field, six independent quantities per point, rendered by the technique of Tensor Patterns. The two orbiting black holes appear as ploughing through the spacetime.

metric tensor field, which also includes the information about the second eigenvector and the shape factor information. Such are depicted in figure 25 using the technique of Tensor Patterns. Linear regions (one dominant eigenvector) appear green, red regions indicate two dominant eigenvectors. The red/green structures are thus similar to figure 23, but here we also see the orientation of the eigenvectors. Visually, the two black holes appear to drag a coma of a planar region behind them and seem to plough through the spacetime, thereby emitting a gravitational wave of highly linear stretching. This visualisation covers a six-dimensional fiber space over a 4-dimensional base space.

4. Data Organisation

In the previous sections, we were considering the data sets as fibers given on a base space, i.e. each point in base space has the same amount of data attached. An unstructured grid or particle set with fields implements this mathematical model most generically. However, it is not desirable to *convert* all structured data types into unstructured ones and loose the additional information hereby. Rather, we need to have the visualisation algorithm to operate directly on the given data, structured or unstructured, in whatever form they exist for a given case.

It is however not possible to employ generic programming here (Veldhuizen 1995)

because the graphics hardware requires a specific layout of the data and the type of the data set is only known at runtime when loading data from within an interactive application. Thus, in order to allow a generic visualisation algorithm to operate *in situ* on the different data types, we need to impose a certain structure on the concept of a *field* over set of points. We identified these four properties that the implementation of a field as a data structure needs to support:

(i) **Hierarchical ordering:** For a certain point in space, there exist multiple data values, one for each *refinement* level. This property describes the topological structure of the base space.

(ii) **Multiple coordinate systems:** One spatial point may have multiple data representations relating to different coordinate systems. This property describes the geometrical structure of the base space.

(iii) **Fragmentation:** Data stem from multiple sources, such as a distributed multiprocess simulation. The field then consists of multiple data blocks, each of them covering a subdomain of the field's base space. Such field fragments may also overlap.

(iv) **Separated Compounds:** A compound data type such as a vector or tensor, may be stored in different data layouts since applications have their own preferences. An array of tensors may also be stored as a tensor of arrays, e.g. `XYZXYZXYZXYZ` as `XXXXYYYYZZZZ`. This property describes the internal structure of the fiber space.

All of these structure components are optional. In the most simple case, a field is just represented by an array of native data types; however, in the most general case (which the visualisation algorithm must always support) the data are distributed over several such property elements and build from many arrays.

In our implementation these data arrays describing the entity of a *field* are organized in a hierarchy of four levels, as illustrated in figure 26. The uppermost level (i) embraces multiple resolutions of the spatial domain; each of these levels may have different topological properties. Each such level may encompass multiple coordinate representations (ii) covering the same topological points. Data per patch may well be distributed over various fragments (iii), whereby fields of multiple components such as vector or tensor fields may be separated into disting arrays itself (iv).

5. Conclusion

In this article we have discussed a systematic approach on how to visualise multiple fields on a spatio-temporal domain, based on the properties of Gaussian footprints representing the various fields. Such properties can be colour, size, or geometric displacement as well as texturing and additional structure of these footprints. This approach is intrinsically independent from the topological structure of the underlying mesh, and an appropriate data structure has been presented that allows to directly represent specific meshes

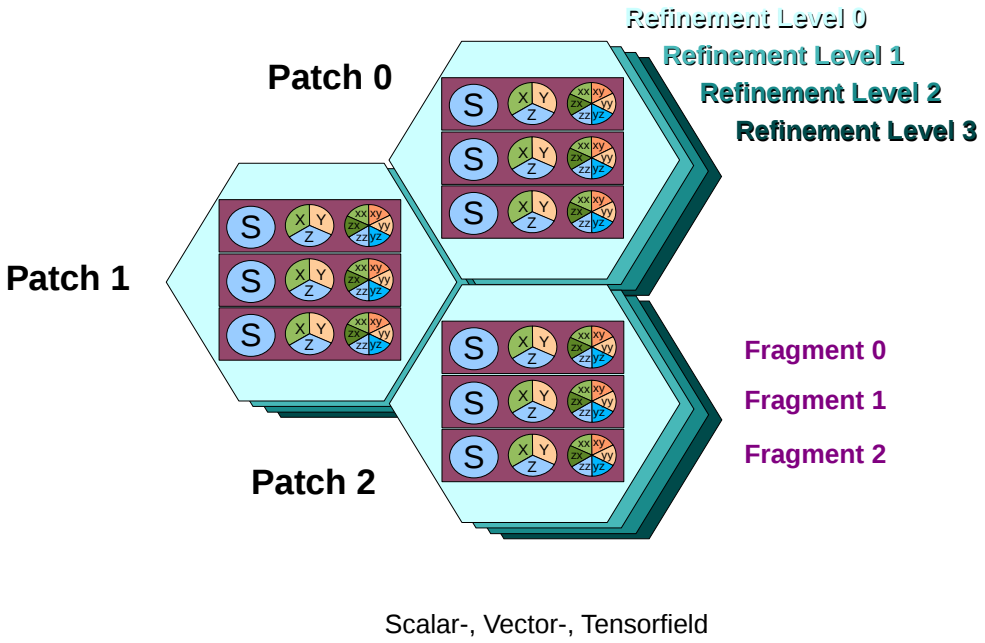

Figure 26. Hierarchical Structure of the data layout of the concept of a *field* in computer memory: (i) Organisation by multiple resolutions for same spatial domain; (ii) multiple coordinate systems covering different spatial domains (arbitrary overlap possible); (iii) fragmentation of fields into blocks (recombination from parallel data sources); (iv) layout of compound fields as components for performance reasons, indicated as S (scalar field), $\{x, y, z\}$ for vector fields and $\{xx, xy, yy, yz, zz, zx\}$ for tensor fields.

without need to convert these data sets. The methods have been demonstrated upon recently generated datasets stemming from practical applications.

6. Acknowledgements

The author thanks Georg Ritter, University of Innsbruck, and Imtiaz Hossain, LSU for their contributions to the article; furthermore Dr. Wolfgang Kapferer, University of Innsbruck, for providing the SPH dataset of the colliding galaxies; Dr. Burkhard Zink, LSU, for providing the multipatch dataset of the neutron star Dr. Erik Schnetter, LSU, for providing the AMR dataset of the merging black holes and his contributions to this article. This research also employed resources of the Center for Computation & Technology at LSU, which is supported by funding from the Louisiana legislature's Information Technology Initiative. Portions of this work were supported by

NSF/EPSCoR Award No. EPS-0701491 (CyberTools).

References

Baker J G, Centrella J, Choi D I, Koppitz M & van Meter J 2006 *Phys. Rev. Lett.* **96**, 111102.

Benger W 2004 Visualization of General Relativistic Tensor Fields via a Fiber Bundle Data Model PhD thesis FU Berlin.

Benger W, Bartsch H, Hege H C, Kitzler H, Shumilina A & Werner A 2006 *International Journal of Neuroscience* **116**(4), pp. 461–514.

Benger W & Hege H C 2004 *in* Erbacher, Chen, Roberts, Gröhn & Börner, eds, 'Conference on Visualization and Data Analysis 2004' Proceedings of SPIE Vol. #5295 pp. 151–162. IS&T/SPIE Electronic Imaging Symposium in San Jose, CA.
`http://www.zib.de/visual/publications/sources/VDA2004.pdf`

Berger M J & Oliger J 1984 *J. Comput. Phys.* **53**, 484–512.

Berti G 2000 Generic Software Components for Scientific Computing PhD thesis University of Cottbus.
`http://www.math.tu-cottbus.de/~berti/diss/`.

Butler D M & Bryson S 1992 *Computers in Physics* **6**, 576–584.

Butler D M & Pendley M H 1989 *Computers in Physics* **3**(5), 45–51.

Campanelli M, Lousto C O, Marronetti P & Zlochower Y 2006 *Phys. Rev. Lett.* **96**, 111101.

Cook L & Matarazzo C 1998 *in* '1998 Nuclear Explosives Development Conference, Las Vegas, NV'.

Crawfis R A & Max N 1993*a* pp. 91–98.

Crawfis R & Max N 1993*b* *in* Nielson & Bergeron, eds, 'Visualization '93, San Jose' IEEE CS Press pp. 261–266.

Diener P, Herrmann F, Pollney D, Schnetter E, Seidel E, Takahashi R, Thornburg J & Ventrella J 2006 *Phys. Rev. Lett.* **96**, 121101.
`http://arxiv.org/abs/gr-qc/0512108`

GEO600 2008. GEO 600.
`http://www.geo600.uni-hannover.de/`

Kähler R 2005 Accelerated Volume Rendering on Structured Adaptive Meshes Ph.d. thesis Freie Universität Berlin, Fachbereich Mathematik und Informatik (P. Deuflhard).
`http://www.zib.de/visual/publications/thesis/DissKaehler.pdf`

Kapferer W, Knapp A, Schindler S, Kimeswenger S & van Kampen E 2005 'Star formation rates and mass distributions in interacting galaxies' `http://arxiv.org/abs/astro-ph/0503559v1`.

Kaufman A & Mueller K 2004 Academic Press.
`http://www.cs.sunysb.edu/~mueller/papers/volvisOverview.pdf`

Kindlmann G & Durkin J W 1998 pp. 79–86.
`citeseer.ist.psu.edu/kindlmann98semiautomatic.html`

Kitware 2005 'Visualization toolkit'. Last visited 4/25/2007.
`http://www.kitware.org/`

Kreylos O, K.-L. Ma K L & Hamann B 2000 *in* 'Proceedings of the Workshop on Computer Graphics and Virtual Reality, 2000 International Computer Symposium'.

Laidlaw D, Ahrens E, Kremers D, Avalos M J, Jacobs R E & Readhead C 1998 *in* 'Proc.Viz.'98' pp. 127–134.
`http://www.cs.brown.edu/people/dhl/pdf/vis98dti.pdf`

LIGO 2008. LIGO: Laser Interferometer Gravitational Wave Observatory.
`http://www.ligo.caltech.edu/`

Schnetter E, Hawley S H & Hawke I 2004 *Class. Quantum Grav.* **21**(6), 1465–1488.

Springel V 2005 *Mon.Not.Roy.Astron.Soc.* **364**, 1105–1134.
`http://www.mpa-garching.mpg.de/gadget`

Stalling D, Westerhoff M & Hege H C 2005 *in* C. R Johnson & C. D Hansen, eds, 'Visualization

Handbook' Academic Press.
`http://www.amiravis.com/`

Teufel H J & Wehrhahn C 2000 *J. Opt. Soc. Am. A* **17**(6), 994–1006.
`http://josaa.osa.org/abstract.cfm?URI=josaa-17-6-994`

The HDF5 Group 2008 'Hierarchical data format version 5' `http://hdf5.ncsa.uiuc.edu/hdf5/`.

Treinish L A 1997 'Data explorer data model' `http://www.research.ibm.com/people/l/lloydt/dm/dx/dx_dm.htm`.

Veldhuizen T 1995 *C++ Report* **7**(4), 36–43. Reprinted in C++ Gems, ed. Stanley Lippman.

Weber G H, Beckner V E, Childs H, Ligocki T J, Miller M C, Straalen B V & Bethel E W 2007 *in* '4^{th} High-End Visualization Workshop, Obergurgl, Tyrol, Austria, June 18-21, 2007' Berlin, Lehmanns Media-LOB.de pp. 12–25.

Weber G H, Kreylos O, Ligocki T J, Shalf J M, Hagen H, Hamann B, Joy K I & Ma K L 2001 *in* 'Vision, Modeling, and Visualization 2001' Akademische Verlagsgesellschaft Aka GmbH and IOS Press BV Berlin, Germany and Amsterdam, Netherlands pp. 121–128, 522.

Westin C, Peled S, Gudbjartsson H, Kikinis R & Jolesz F 1997 *in* 'Proceedings of ISMRM, Fifth Meeting, Vancouver, Canada' p. 1742.
`http://spl.bwh.harvard.edu:8000/pages/ppl/westin/papers/smr97/htmlversion.html`

Westover L 1990 *in* 'Proceedings of the 17th annual conference on Computer graphics and interactive techniques' ACM Press, NY, USA pp. 367–376.

Zink B, Schnetter E & Tiglio M 2007 'Multi-patch methods in general relativistic astrophysics - I. Hydrodynamical flows on fixed backgrounds' `http://arxiv.org/abs/0712.0353`.