

Visualizing Complicated Dynamics

D. Buerkle, M. Dellnitz, O. Junge, M. Rumpf, M. Spielberg

Abstract

The temporal evolution of real world systems can mathematically be described by dynamical systems. Global, topological information on their long term behavior is given by corresponding invariant sets, which typically have a very complex structure. The rendering of a single, long trajectory - the standard visualization approach - only very partially shows the rich structure with the frequently chaotic dynamical behavior. We describe a new, set oriented and hierarchical approach for the computation and visualization of invariant sets and accompanying probability measures, which give further information on the dynamical system.

Our approach comes along with a stable, dimension independent approximation scheme, with tailored data structures of compressed pyramidal type, and an efficient volume rendering method. Furthermore a dense coverage of the invariant sets with many short transparent streamlines arranged randomly allows a combined visualization of the global structure and the underlying local dynamics.

1 Introduction

The temporal evolution of a real world system can mathematically be described by a *dynamical system*. If time is assumed to go on continuously then this system is frequently given by an ordinary differential equation of the form

$$\frac{dx}{dt}(t) = g(x), \quad (1)$$

where $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$. As an interesting first example let us consider *Chua's circuit*

$$\begin{aligned} \dot{x} &= \alpha(y - m_0x - \frac{1}{3}m_1x^3) \\ \dot{y} &= x - y + z \\ \dot{z} &= -\beta y, \end{aligned} \quad (2)$$

which is a mathematical model describing the dynamical behavior of a certain electrical circuit (see [14]) (Here α , β , m_0 and m_1 are real parameters related to the different components of the circuit). On the other hand, if we consider the evolution process to be regular but look at it just at single instances of time, then the mathematical model is a discrete dynamical system of the form

$$x_{j+1} = f(x_j), \quad j = 0, 1, \dots, \quad (3)$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$. Observe that this type of dynamical system naturally arises when an ordinary differential equation is integrated by an explicit numerical scheme. In both cases – in continuous

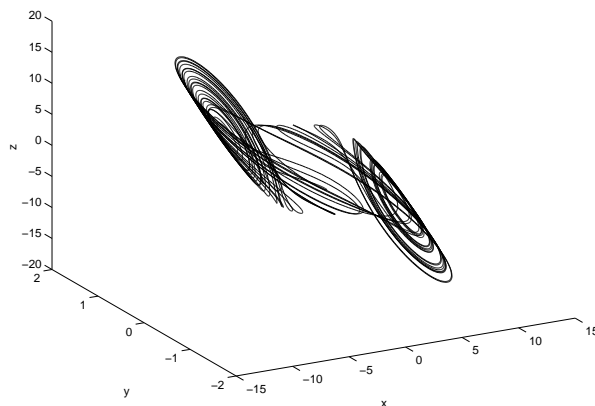


Figure 1: Approximation of an invariant set of Chua's circuit by direct simulation.

and in discrete time – the purpose of a numerical study of the dynamical system is to extract valuable information on its long term dynamical behavior.

Topological information on the long term behavior of (3) is given by invariant sets: a set $A \subset \mathbb{R}^n$ is *invariant* if

$$f(A) = A.$$

For instance, A could simply consist of a point, i.e. a *fixed point*, but it could also have a complicated structure. In the latter case the underlying dynamical behavior is typically *chaotic*. Up to now the standard method to approximate “chaotic” invariant sets is to compute a single solution of the underlying dynamical system for a long period of time and then to display this solution in state space. In Figure 1 we show such a result for Chua's circuit (2). Here we get a first glance on the complex dynamical behavior. However, several important questions cannot be answered by examining Figure 1:

- Will a typical trajectory eventually enter some of the visible “holes” ?
- How likely is it to observe a typical trajectory in a certain subset in state space?
- Are there any regions where a typical trajectory will stay for a longer period of time before entering other regions?

Fig. 2 gives an impression of a new approach which could be used to close this gap. It allows an intuitive perception of the complete, rich geometry of the same invariant set and is build up by a dense coverage with short streamlines. They are shaded with respect to

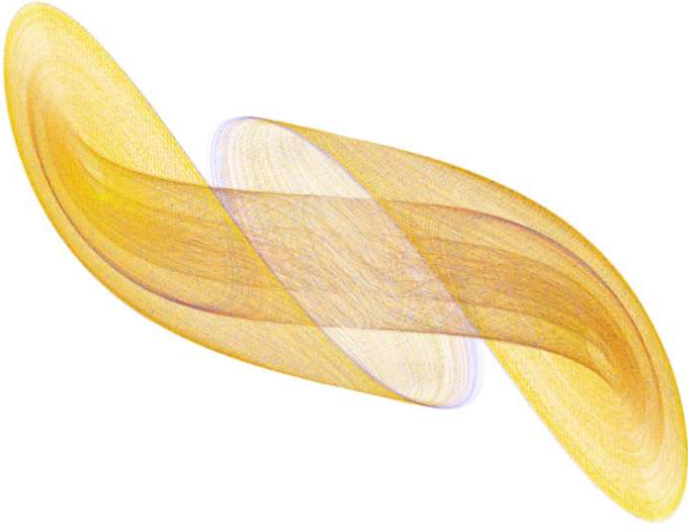


Figure 2: A dense coverage of Chua's circuit with transparent shaded streamlines.

the surface characteristics, and color represents the probability to meet a particle on a “typical” trajectory. Furthermore compare the shaded volume rendering in Fig. 10, where color represents two almost invariant sets.

The purpose of this article is to present in detail such novel visualization techniques based on recent numerical approximation methods. The *set oriented*, hierarchical numerical algorithms are presented in Section 2. With the help of these algorithms the invariant set is approximated by an *exterior box covering* rather than by *interior testing with single trajectories*. Thus “holes” as in Figure 1 cannot occur. Moreover, the desired information on the dynamical behavior can be determined efficiently. The numerical algorithms are similar to the well known *cell mapping techniques* [13]. However, there is a crucial difference between the two: the numerical effort essentially depends on the complexity of the underlying dynamics and not on the dimension of state space.

The objects that we are going to compute and visualize are (*almost*) *invariant sets*, *invariant manifolds* and *invariant measures*. The invariant manifolds of an invariant set consist of those points in state space which approach the set asymptotically in forward resp. backward time: for an invariant set A define

$$\begin{aligned} W^s(A) &= \{x \in \mathbb{R}^n : f^j(x) \rightarrow A \text{ for } j \rightarrow +\infty\}, \\ W^u(A) &= \{x \in \mathbb{R}^n : f^j(x) \rightarrow A \text{ for } j \rightarrow -\infty\}. \end{aligned} \quad (4)$$

Then, under certain additional hypotheses on the invariant set A (see [12, 8]) these objects are indeed manifolds and are called the *stable* resp. the *unstable manifold* of A . Obviously these manifolds are invariant sets.

The statistical information on the dynamical behavior on an invariant set is given by a corresponding invariant measure: a probability measure μ is *invariant* if $\mu(B) = \mu(f^{-1}(B))$ for all (measurable) subsets $B \subset \mathbb{R}^n$. For instance, if A is a fixed point p then the Dirac measure δ_p is an invariant measure. We will present visualization techniques for the “physically relevant” invariant measure, that is,

the measure which provides the statistics for typical solutions of the underlying system. The notion of an *almost* invariant set will be further explained in Section 2.

In what follows our main objective is to explain the robust and efficient algorithms for the computation and visualization of invariant sets. To ensure efficiency the set covering is constructed as a binary, hierarchical tree of boxes. Thereby, especially very fine approximations of the complicated set with millions of boxes can be handled in compressed pyramidal data structures as already proposed by Gargantini [10] or Tamminen and Samet [25]. In the numerical algorithm the searching for images of boxes under the mapping f can be implemented in a multilevel fashion [20]. Furthermore, these pyramidal data structures support efficient interactive postprocessing techniques related to approaches like the one presented by Ghavannia and Yang [11].

Besides these general data handling considerations appropriate concrete visualizations methods are required to ensure a intuitive understandable visual impression of the considered dynamical system. At first, volume rendering techniques as outlined by Max [18], based on transparent box projections, introduced by Shirley and Tuchman [22] and in a hierarchical context by Laur and Hanrahan [16], allow a graphical representation of the approximate invariant sets. Appropriate shading of the volumetric box covering is an important issue to improve the three dimensional perception of the considered sets, which typically approximate locally smooth manifolds. Approximate normal spaces can be defined similar to Nielson's approach [3] in scattered data interpolation. Then shading with respect to these normal spaces is possible, according to the considerations by Banks [1]. A very first version of this volume rendering, without splatting and shading is presented in [4].

Secondly, our intention is once more to incorporate the local dynamics on the invariant set in the visualization. It can be represented by local trajectories of the underlying ODE. In 2D van Wijk [27] applies texture spots to generate trajectory aligned patterns. Cabral and Leedom [2] propose the line integral convolution method to enhance the dense visualization of trajectories. These techniques have been exploited for dynamical systems in 2D, or on slices in 3D by Wegenkittl et al. [29]. Here we will directly tackle the full 3D problem.

Our approach is related to the method of illuminated streamlines introduced by Stalling et al. [23]. Here, a coverage of the frequently lower dimensional invariant manifolds is attained similar to the line art illustration method by Elber [9]. Other somehow related approaches to visualize 3D flow fields by enhancing local trajectories are due to van Wijk [28] and Max et al. [19].

2 The Set Oriented Algorithms

In the following paragraphs we will briefly describe the numerical algorithms used for the computation of the topological and statistical information on the underlying dynamical behavior of system (3).

A Subdivision Algorithm

The central object which is approximated by the subdivision algorithm developed in [6] is the so-called *relative global attractor*,

$$A_Q = \bigcap_{j \geq 0} f^j(Q), \quad (5)$$

where $Q \subset \mathbb{R}^n$ is a compact subset. Roughly speaking, the set A_Q should be viewed as the union of invariant sets inside Q together with their unstable manifolds. In particular, A_Q may contain subsets of Q which cannot be approximated by direct simulation.

The subdivision algorithm for the approximation of A_Q generates a sequence $\mathcal{B}_0, \mathcal{B}_1, \mathcal{B}_2, \dots$ of finite collections of boxes with the property that for all integers k the set

$$A_Q^k = \bigcup_{B \in \mathcal{B}_k} B$$

is a covering of the relative global attractor under consideration. Moreover, the sequence of coverings is constructed in such a way that the diameter of the boxes,

$$\text{diam}(\mathcal{B}_k) = \max_{B \in \mathcal{B}_k} \text{diam}(B)$$

converges to zero for $k \rightarrow \infty$.

Given an initial collection \mathcal{B}_0 , one inductively obtains \mathcal{B}_k from \mathcal{B}_{k-1} for $k = 1, 2, \dots$ in two steps.

(i) *Subdivision*: Construct a new collection $\hat{\mathcal{B}}_k$ such that

$$\bigcup_{B \in \hat{\mathcal{B}}_k} B = \bigcup_{B \in \mathcal{B}_{k-1}} B$$

and

$$\text{diam}(\hat{\mathcal{B}}_k) \leq \theta \text{diam}(\mathcal{B}_{k-1})$$

for some $0 < \theta < 1$.

(ii) *Selection*: Define the new collection \mathcal{B}_k by

$$\mathcal{B}_k = \left\{ B \in \hat{\mathcal{B}}_k \mid \exists \hat{B} \in \hat{\mathcal{B}}_k \ f^{-1}(B) \cap \hat{B} \neq \emptyset \right\}.$$

The following proposition establishes a general convergence property of this algorithm.

PROPOSITION 2.1 ([6]) *Let A_Q be the global attractor relative to the compact set Q , and let \mathcal{B}_0 be a finite collection of closed subsets with $Q_0 = Q$. Then*

$$\lim_{k \rightarrow \infty} h(A_Q, A_Q^k) = 0,$$

where we denote by $h(B, C)$ the usual Hausdorff distance between two compact subsets $B, C \subset \mathbb{R}^n$.

In our concrete implementation the boxes are generalized rectangles (cf. Section 3).

Based on the subdivision algorithm there also exists a numerical globalization technique producing coverings of invariant manifolds of invariant sets (see (4)). The idea is to compute a local covering

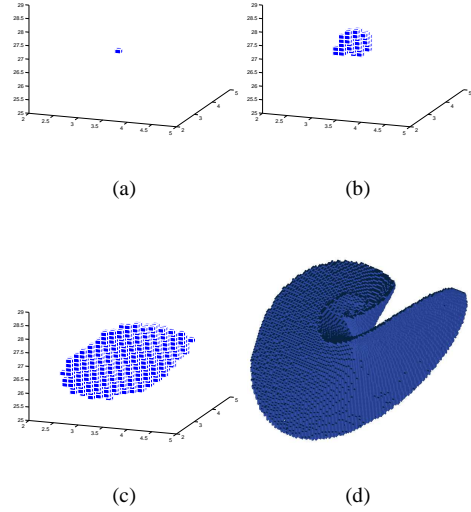


Figure 3: Covering of a two-dimensional unstable manifold of a steady state solution p in the Lorenz system. (a) Covering of $W_{\text{loc}}^u(p)$; (b), (c) two steps in the globalization of $W_{\text{loc}}^u(p)$; (d) the whole manifold.

of these manifolds by the subdivision procedure and then to globalize this covering by repeated application of the dynamical system f . A detailed discussion of this method can be found in [5].

We will now illustrate this procedure by an example. In Figure 3 the growth process leading to a covering of a two-dimensional unstable manifold of a steady state solution in the *Lorenz system*

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy \end{aligned} \quad (6)$$

is shown. For the computations we have set

$$\sigma = 10, \quad \rho = 28 \quad \text{and} \quad \beta = 0.4,$$

and the steady state solution is $p \approx (3.29, 3.29, 27)$. It is well known that p possesses a two-dimensional unstable manifold $W^u(p)$.

Locally Flat Attractors

Our set oriented algorithm is not restricted to approximations of attracting sets which are smooth submanifolds of \mathbb{R}^n . The attractor under consideration may have a Hausdorff dimension which is not an integer (cf. the Lorenz attractor in Fig. 8, which is of Hausdorff dimension between two and three). Nevertheless, frequently attractors are contained in the closure of unstable manifolds which are locally m dimensional surfaces within \mathbb{R}^n . Unfortunately, this surface structure is hidden in our discrete box approach. In terms of a surface interpretation the fundamental question is how to define a tangent space, or, equivalently, how to give a suitable definition of normals. Here we apply a method related to Nielson's approach in the interpolation of scattered data [3].

For given ε we consider the neighbourhood $U_\varepsilon(c_B)$ of the center point c_B of every box $B \in \mathcal{B}_k$ (The distance is measured in the maximum norm). Then we define the local center of gravity \bar{x}_B and the first momentum matrix S_B by

$$\bar{x}_B = \int_{U_\varepsilon(c_B) \cap A_Q^k} dx, \quad S_B = \frac{3}{\varepsilon^2} \int_{U_\varepsilon(c_B) \cap A_Q^k} (x - \bar{x}_B)(x - \bar{x}_B)^T dx.$$

The scaling of S_B obviously guarantees that $S_B = Id$, if the invariant set completely covers $U_\varepsilon(c_B)$. Concerning the implementation we can avoid the exact evaluation of the integrals by approximation with a counting measure over box centers. We thereby take into account values of ε which are multiples of the length of box edges, e. g. in the applications we consider a factor of 10.

The momentum matrix S_B is symmetric. Thus, there exists an orthonormal system of eigenvectors v_1, \dots, v_n and corresponding real eigenvalues $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$. By construction the approximate set A_Q^k is locally more extended in directions of eigenvectors with relatively large eigenvalues λ_j and vice versa. If the actual invariant set A_Q is locally an m dimensional surface, then it is reasonable to assume that A_Q^k reflects this property in the sense that there are $(n - m)$ small eigenvalues, i. e.

$$\lambda_1 \leq \dots \leq \lambda_{n-m} \ll \lambda_{n-m+1} \leq \dots \leq \lambda_n.$$

We make use of this fact and require in the algorithm that $\frac{\lambda_{n-m}}{\lambda_{n-m+1}} < \delta$ for a small constant δ , in our case 0.1. Then the eigenvectors v_1, \dots, v_{n-m} are interpreted as approximate normals and v_{n-m+1}, \dots, v_n as approximate tangent vectors. Concerning the visualization, especially for $n = 3$ and $m = 2$ the definition of a normal allows an appropriate shading and thereby supports the visual reception of complicated invariant manifolds (cf. Fig. 8). This can easily be generalized in the sense of Banks [1]. In the three dimensional case S_B is a 3×3 -matrix and therefore its eigenvalues can be computed by a direct approach.

The Computation of Invariant Measures

Once a box covering \mathcal{B} of the attractor A_C has been computed, we can approximate the statistics of the dynamics on A_C by the computation of a corresponding natural invariant measure. We now briefly describe how this measure can be approximated. For the precise statements on the convergence of this method see e.g. [15, 7].

Let $B_i \in \mathcal{B}$, $i = 1, \dots, N$, denote the boxes in the covering obtained after a certain number of steps in the subdivision algorithm or the globalization technique for invariant manifolds. Then we compute the transition probabilities

$$p_{ij} = \frac{m(f^{-1}(B_i) \cap B_j)}{m(B_j)}, \quad i, j = 1, \dots, N,$$

where m denotes Lebesgue measure. Let $P = (p_{ij})$ be the corresponding transition matrix. Then P is a stochastic matrix and a fixed point $u = Pu$ of P gives the desired approximation to an invariant measure μ of f . This fix point can be identified by computing the eigenvector for the eigenvalue 1 (cf. Section 3). This

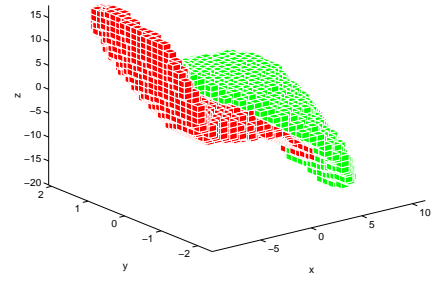


Figure 4: A coarse box coverings of two almost invariant sets in Chua's circuit (red, respectively green).

natural invariant measure μ provides the information where, on average, typical trajectories will be observed in the given system.

However, there is additional statistical information on the underlying dynamical behavior which cannot be described by the invariant measure itself. For instance, it may be of interest to know the average time for which a typical trajectory remains inside a certain region R in phase space. This period of time is related to the number

$$\delta(R) = \frac{\mu(f^{-1}(R) \cap R)}{\mu(R)},$$

since $\delta(R)$ can be viewed as the probability that points in R are mapped into R under f . In particular, if R is an invariant set, that is $f^{-1}(R) = R$, then $\delta(R) = 1$.

In [7] it has been shown that $\delta(R)$ is strongly related to a certain part of the spectrum of P , and again solving an eigenvalue problem we use this information in our algorithms for the identification of *almost invariant sets*. These are sets where typical trajectories stay for a long period of time. As an example we show in Figure 4 two box coverings of almost invariant sets for Chua's circuit (2). In the computations we have set

$$\alpha = 18, \beta = 33, m_0 = -0.2, m_1 = 0.01.$$

3 Implementational aspects

So far, we have introduced a new robust set oriented approach to handle complicated invariant sets from discrete dynamical systems. The efficiency of the subdivision algorithm and the visualization methods significantly depends on suitable hierarchical data structures to handle the corresponding box hierarchies. We are thereby able to ensure sparse storage requirements in particular for very large box collections which give a suitable fine approximation of the considered invariant sets. Let us now discuss the concrete implementation of such a hierarchical data structure and the corresponding subdivision algorithm.

We assume the boxes $B \in \mathcal{B}_k$ to be generalized rectangles in \mathbb{R}^n which are tensor products of intervals in the coordinate directions. The subdivision algorithm is started with a single rectangle $B_0 = Q$, i.e. $\mathcal{B}_0 = \{Q\}$. At first we have to choose an appropriate box refinement rule. In two and three dimensions a quadtree, respectively an

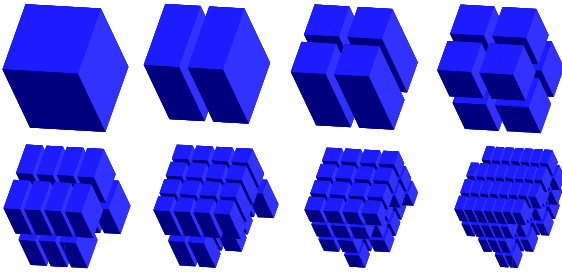


Figure 5: The box collections \mathcal{B}_k , $k = 0, \dots, 7$.

octree, would be a possible choice. But in higher dimensions (cf. the applications in Section 5) this turns out not to be very hand-some. To keep the algorithm dimension independent we consider a binary tree of boxes instead. It comes along with the successive bisection of boxes cycling over the coordinate directions (cf. Fig. 5). Let us denote a n dimensional box by $B = [l, u] = [l_0, u_0] \times \dots \times [l_{n-1}, u_{n-1}]$. Such a box $B = [l, u] \in \mathcal{B}_k$ is subdivided in the coordinate direction $s = k \bmod n$ into two child boxes B^0, B^1 , where

$$B^i := [l^i, u^i] \quad (7)$$

with $l_j^i = l_j, u_j^i = u_j$ for $j \neq s$, and $l_s^0 = l_s, u_s^0 = \frac{l_s + u_s}{2}$ respectively $l_s^1 = \frac{l_s + u_s}{2}, u_s^1 = u_s$. By evaluating the box diameter $h_k := \text{diam}(B)$ we obtain

$$h_k = \text{diam}(B_{k \bmod n}) 2^{-\frac{k}{n}}.$$

Note that each \mathcal{B}_k and the corresponding covering A_Q^k are completely determined by the tree structure and the initial box Q . In any depth first traversal of the tree complete information on the boxes can be generated at runtime according to equation 7.

In case of a smooth m dimensional invariant set A_Q the corresponding box covering A_Q^k is typically of uniformly bounded thickness, measured in terms of boxes for increasing level k . This leads to $O(h_k^{-m} |\log h_k|)$ boxes in a k level box hierarchy with fine grid granularity h_k and clearly indicates the benefits of the hierarchical data structure in handling these objects compared to an approach based on full grids with the same grid size. The following tables lists the true number of boxes for the Lorenz system, respectively Chua's circuit and different levels k .

k	21	24	27	30
Lorenz ($\beta = 0.4$)	31074	124084	497740	1998566
Chua's circuit	62288	293992	1465152	-

Here, for instance $k = 30$ corresponds to full grids of size 1024^3 . Based on these binary tree structures the abstract subdivision introduced in Section 2 can now easily be implemented:

- Starting with the initial covering \mathcal{B}_0 which consists of the single box $B_0 = Q$, we add in step k a finer layer $\hat{\mathcal{B}}_k$ to the hierarchy: For each box $B \in \mathcal{B}_{k-1}$ the child boxes B^0, B^1 are generated as candidate cells for the refined approximate set. This can be done in one depth first traversal of the current tree structure.
- Then for each box B in $\hat{\mathcal{B}}_k$ we compute $f(B)$ and check which boxes in $\hat{\mathcal{B}}_k$ are intersected by this box image. For general f

this is a difficult task, therefore we furthermore discretize this step. A set of test points is selected on each box B . For each test point x we compute the image $f(x)$ and search for a box in $\hat{\mathcal{B}}_k$ containing $f(x)$. If such a box is found it is marked.

- All unmarked boxes in $\hat{\mathcal{B}}_k$ are deleted. This finally results in the new, refined covering \mathcal{B}_k of the invariant set.

The search for a box containing a distinct point can easily be implemented recursively on the box hierarchy starting at the root box B_0 , where the cost for this fundamental routine increases linearly in k .

Invoking the test points forces us to ask for the robustness of the algorithm. It is no longer ruled out that we might skip boxes although they intersect the invariant set. Nevertheless for a sufficiently large set of test points the number of “lost” boxes can be kept small. Alternatively, if f is Lipschitz continuous the choice of test points and the marking of boxes can be made rigorous in the sense that the computed covering really covers the considered set.

All the set oriented algorithms which are based on the subdivision strategy use the same data structure and essentially the same technique of mapping test points and marking (and inserting) boxes in the hierarchy. For the solution of the eigenvalue problems we use the software package ARPACK ([24, 17]), which implements an Arnoldi method for the computation of a few eigenvalues of a sparse matrix. Hence, we obtain eigenvectors, the components of which correspond to boxes.

Besides an efficient hierarchical implementation of the algorithm the hierarchical data structures also allow an efficient data compression. If we solely consider the geometry of an approximate invariant set A_Q^k the complete tree can be stored as a sequence of bits. Here a value 1 indicates an existing box, whereas 0 marks a nonexisting child. These bits have to be ordered corresponding to a prefix traversal of the binary tree.

For the storage of the computed statistical information, say an invariant measure μ , we use one byte per box, consider a logarithmical scale, and map $\log(\mu)$ from the interval $[\min_{B \in \mathcal{B}_k} \log(\mu(B)), \max_{B \in \mathcal{B}_k} \log(\mu(B))]$ onto some integer M from the integer interval $[1, 255]$. The value 0 may indicate a skipped box.

Especially in three dimensions and in the case of a locally flat invariant manifold we have introduced normals which can be used for shading purposes. We decided to code normals in 3 bytes N_1, N_2, N_3 , where each byte is interpreted as a *signed char* with six significant bits and the actual normal n can be retrieved by the relation $n_i = 0.02 N_i$. For a shading calculation this resolution proved to be sufficient. Two of the remaining unused bits S_1, S_2, S_3 in these bytes can be used as flags indicating the existence of the child boxes. So far at most 4 byte are required to store a single box on disk. The visualization algorithms to be presented in the subsequent sections require a back to front sorting of boxes in the binary tree. We can support this with an expansion of the data structures in memory. For each box $B \in \mathcal{B}_j$ with $j < k$ and two existing children we add a reference to the second child box B^1 , whereas the first child box is stored directly next to the current box (cf. Fig. 6). We are thus able to select any sorted ordering of the child boxes during a depth first traversal in the visualization. If we consider a 4 byte integer reference to the second child we end up with at most

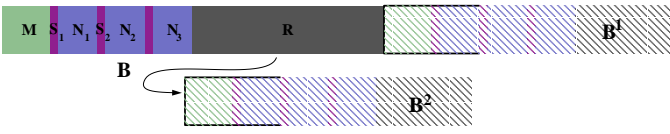


Figure 6: A sketch of the storage scheme for a single box in 3D and its two child boxes

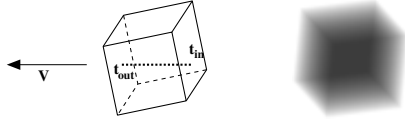


Figure 7: A ray intersecting a box in 3D (left) and a resulting splat (right).

8 byte for an existing box in the box hierarchy up to second finest box level, where on the finest box level 4 byte are still sufficient.

4 Visualization approach

In what follows we will present two different visualization methods. At first, a volume rendering enlightens the geometry of the invariant set under consideration. Furthermore, a dense streamline coverage method shows in addition the local dynamics (cf. the motivating image already in Section 1). We here restrict ourselves to the three dimensional case. This can be straightforwardly expanded to projections of higher dimensional invariant sets in phase space to 3 dimensional state space (cf. Section 5).

Volume Rendering

One appropriate way to visualize the complicated structure of the invariant sets is to render them as transparent volumetric object. In the typical case of relatively thin approximate set A_Q^k our pyramidal data structures are well suited for fast volume rendering applying appropriate splatting techniques [22] on the boxes in the hierarchical tree. Color is used to represent the invariant measure or in case of almost invariant sets to separate the distinct regions. We restrict ourselves to orthogonal projection and proceed as follows:

At first, for given normalized viewing direction V and box level k we generate a single splat at a sufficiently fine resolution for a reference box \hat{B} on level $k \bmod 3$. This splat will represent the box opacity $\alpha_{\hat{B}}$ with respect to a constant predefined density ρ on the box. I. e. for a ray which intersects \hat{B} in the direction V with an entry point t_{in} and an exit point t_{out} (cf. Fig. 7). we obtain an opacity value $\alpha_{\hat{B}} = 1 - e^{-\rho(t_{out} - t_{in})}$.

In a back to front traversal of the hierarchical tree of boxes we use this splat to generate a transparent volume projection of A_Q^k (cf. Fig. 8). Thereby, for each box $B \in \mathcal{B}_k$ the old image intensity I^- is updated to a new intensity I^+ according to the standard convex combination formula [18]

$$I^+ = (1 - \alpha_{\hat{B}}) I^- + \alpha_{\hat{B}} C(\mu(B))$$

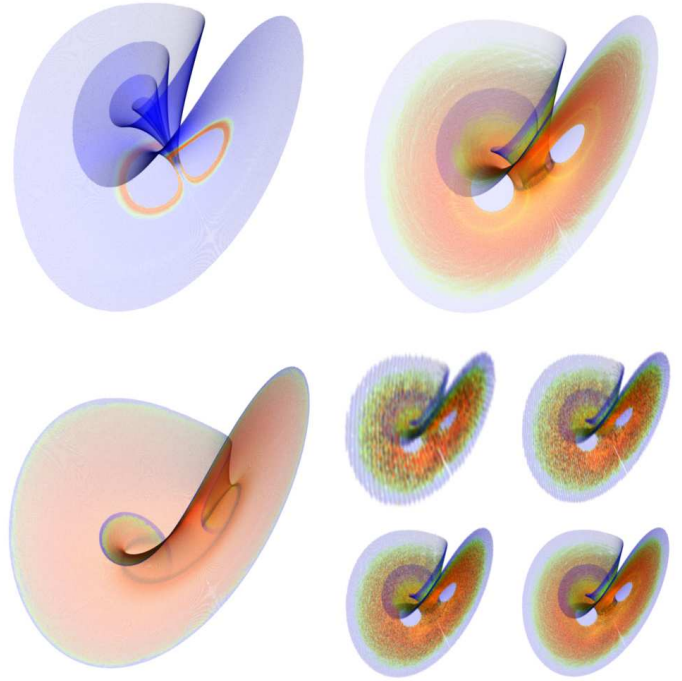


Figure 8: Three members of the Lorenz family are visualized using volume rendering for parameters $\beta = 0.4, 1.2, 2.6$ (left to right, top to bottom). In the lower right corner different levels of detail (15, 18, 21, and 24) are considered. For $\beta = 0.4$ trajectories rapidly approach a double periodic orbit, here indicated by the color concentration. The invariant set for $\beta = 2.6$ is a closed approximation of the famous Lorenz attractor, with an invariant measure equally spread all over the set. This spreading is a clear indication of the underlying chaotic behaviour. Through out the whole sequence we realize a geometric change, but even more dominant this successive spreading of the measure.

which is frequently supported by hardware, for instance using the splat as a transparent texture. The color $C(\mu(B))$ is chosen according to the measure value $\mu(B)$ on the box B . Because the measure typically varies over many orders of magnitude we use a logarithmic scale for the invariant measure, i. e. retrieve the already compressed logarithmic measure values from the data base described in Section 3 (cf. Fig. 9 for a corresponding color ramp on the interval $[0, 1]$). Always taking into account the same coarse reference box leads to a transparency of A_Q^k which is approximately constant for varying level k . In fact for interesting applications the thickness measured in boxes is of the same size. Mathematically this is related to the hyperbolicity of these dynamical systems on the considered invariant sets.



Figure 9: The color ramp corresponding to the visualization of the measure.



Figure 10: Volume rendering of Chua's circuit with different coloring for two almost invariant subsets. The red, respectively blue coloring indicates subsets in which trajectories stay for a very long time until they turn over to the other subset.

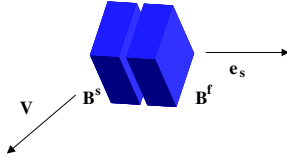


Figure 11: Depending on the viewing direction V child boxes are sorted in the back to front tree traversal

The rendering approach can be generalized, if we consider opacities which depend on the invariant measure μ and take into account approximate box normals n_B for shading purposes. Thus, we obtain a modified update formula

$$I^+ = (1 - \alpha_B(\mu(B))) I^- + \alpha_B(\mu(B)) C(\mu(B), n_B).$$

Here $\alpha_B(\mu(B))$ is any suitable monotone μ dependent scaling of α_B (boxes with smaller measure value are more transparent than those with larger measure value) and the color

$$C(\mu(B), n_B) = C(\mu(B)) + C(n_B)$$

contains an additional component $C(n_B)$ due to a Goraud shading with respect to the normal n_B and given light sources illuminating the invariant manifold. Concerning the implementation we make use of the logarithmic 8 bit quantization of the measure. The splat is copied 256 times and the alpha values of each copy are scaled according to the discrete measure value. Furthermore, we add the appropriate constant color component $C(\mu(B))$. Finally in the back to front traversal of the hierarchical tree, we straightforwardly render Goraud shaded texture splats (cf. Fig. 8).

To conclude, we will now explain the sorting of B_k with respect to the viewing direction. This sorting obviously depends solely on

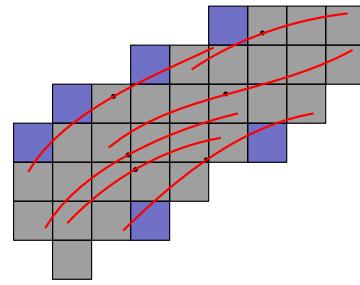


Figure 12: A 2D box collection with streamlines is sketched. The required ghost cells are drawn in blue.

the viewing direction and the bisection direction. For the subdivision direction $s = k \bmod n$ the sign of the s th component of the viewing vector V decides in advance which child to process first (cf. Fig. 11). Defining

$$\text{first}(V, s) = \begin{cases} 0; & V_s > 0 \\ 1; & \text{else} \end{cases} \quad \text{second}(V, s) = \begin{cases} 1; & V_s > 0 \\ 0; & \text{else} \end{cases}$$

we can sketch the pure recursive sorting procedure in the following pseudo code acting on boxes B on level k :

```

BackToFront( $B, k$ ) {
   $s = k \bmod n$ ;
   $f = \text{first}(V, s)$ ;  $s = \text{Second}(V, s)$ ;
  if ( $B^f$  exists) BackToFront( $B^f, k + 1$ );
  if ( $B^s$  exists) BackToFront( $B^s, k + 1$ );
}

```

Dense Covering with Streamlines

The volume rendering approach described so far enables us to visualize the complete geometry together with the invariant measure, respectively the splitting into almost invariant subsets. An impression of the local dynamics on the invariant set A_Q , i. e. the direction and velocity of the continuous flow according to the underlying ODE are still missing in this graphical representation. Streamlines are suitable tools to visualize such information. We will now describe an algorithm which generates a coverage of A_Q^k with streamlines at a prescribed density in a preprocessing step. Then, for the later on interactive rendering we use transparent illuminated streamlines similar to those described by Stalling et al. [23] and color them according to the invariant measure. Except for our case we make use of the approximate surface normals and therefore shade the individual streamlines with respect to these normals. Hence, we interpret them as thin strips attached to the surface similar to van Wijk's oriented surface particle approach [28], whereas in [23] the streamlines appear as thin tubes. We thus ensure the graphical representation of the *global geometry* and the *local dynamics* of the dynamical system at the same time while still retaining the *surface type appearance*.

Our coverage will be of equal density all over A_Q^k in the sense that the ratio $\gamma(B) := L(B)/\text{vol}(B)$ of the sum $L(B)$ of the length of streamline segments in the boxes B of the binary tree and the local volume $\text{vol}(B) = \int_{B \cap A_Q^k} dx$ is balanced. We achieve this by an in-

terative insertion process of streamlines of fixed length 2δ . We successively select starting positions x_0 , compute streamline segments $x : [-\delta, \delta] \rightarrow \mathbb{R}^n$ as numerical solutions of the ODE problems

$$\dot{x} = g(x); \quad x(0) = x_0$$

and distribute the local segments onto the corresponding boxes. Simultaneously we update the current densities $\gamma(B)$ on the involved boxes.

A more detailed description of how new starting positions x_0 are selected is given by the following recursive procedure on the binary tree, which in the first instance allocates a box to release x_0 with respect to the current γ distribution.

```

SelectBox( $B$ ) {
  if both child boxes  $B^0, B^1$  exist {
     $\theta = \frac{\gamma(B^0)}{\gamma(B^0) + \gamma(B^1)}$ ;
    compute a random number  $r \in [0, 1]$ ;
    if ( $r < \theta$ ) SelectBox( $B^1$ );
    else SelectBox( $B^0$ );
  }
  else if the child box  $B^0$  exists
    SelectBox( $B^0$ );
  else if the child box  $B^1$  exists
    SelectBox( $B^1$ );
}

```

Here we assume $\frac{0}{0} = 0$ in case of still empty boxes. Let us emphasize that this recursive call may be repeated on failure, i.e. if a box without child boxes is reached. Subsequently, in each selected box $B \in \mathcal{B}_k$ a random starting position x_0 is chosen.

We apply a 4th order adaptive Runge Kutta method for the integration of the ODE, polygonalize the resulting curves, and clip them at box boundaries (cf. Fig. 12). Then the resulting local segments are stored in lists linked to the boxes. During the traversal we update the densities $\gamma(B)$ for all boxes in the binary tree.

Afterwards, an interactive rendering will refer to the line segment lists in depth first traversal of the tree while drawing a coverage of illuminated transparent streamlines. Again a proceeding from back to front on the box hierarchy supports a sorting necessary for the correct handling of the transparency. Thereby, we do not obtain a strict sorting of all streamline segments. In a single box the corresponding lines are still unsorted. Nevertheless, from our experience, errors resulting from these restrictions are not visible on the final images. We thus avoid a time consuming additional local sorting and the possibly necessary cutting operations in order to ensure an interactive performance of the rendering. In addition, the local dynamics on the invariant set can be animated cyclically shifting highlights over the streamlines as it has been proposed by Zöckler et al. [23] for whose illuminated streamlines. Fig. 13 and 2 show results for Lorenz system, and Chua's circuit. They correspond to the volume rendering in Fig. 8, respectively Fig. 10.

Let us finally discuss a peculiar difficulty. The streamlines may leave the box covering, because they are not released on the exact invariant set. In this case ghost boxes are added to the hierarchy and portions of the streamlines outside the original boxes are stored in them (cf. Fig. 12). These ghost cells are not considered in the subsequent selection steps.

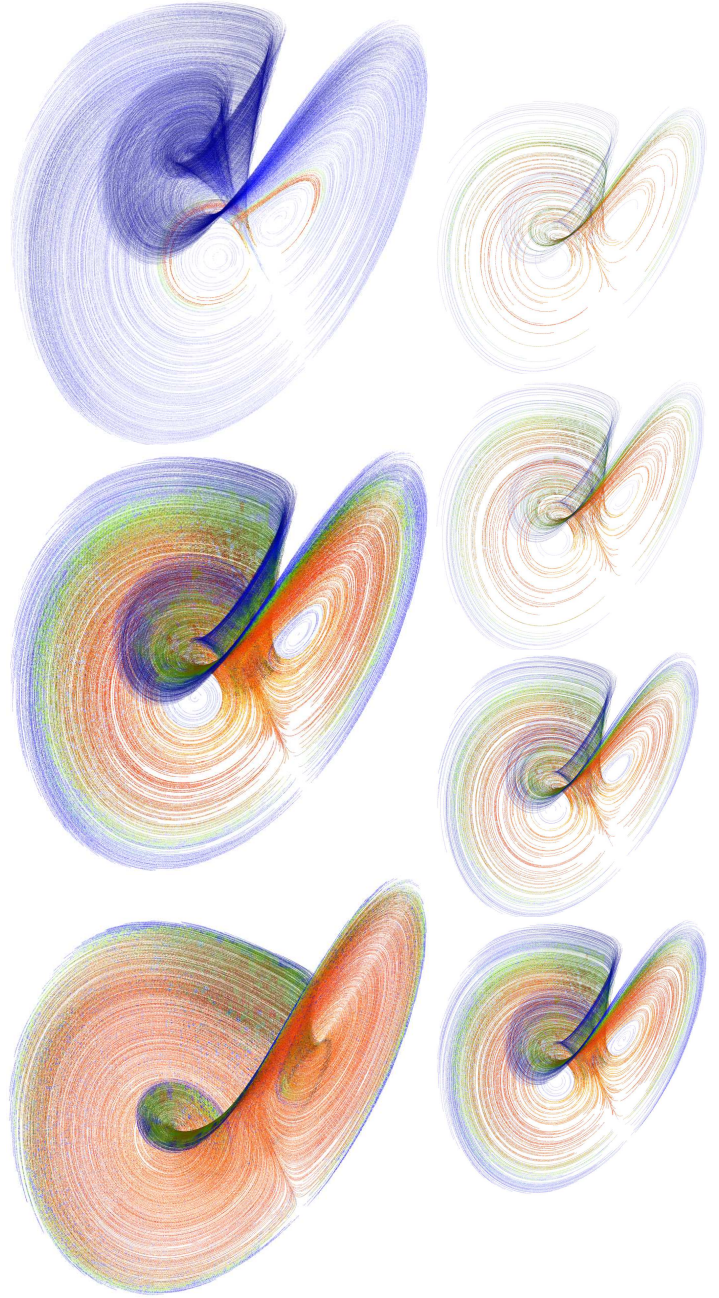


Figure 13: Invariant sets in the Lorenz family are visualized using a coverage with shaded streamlines. On the right streamline coverings of different density are depicted.

The implementation of the covering algorithm requires additional memory, especially for the list of streamline segments. This memory requirement grows exponentially for increasing level k . In the application it turned out that a sorting of the streamline segments is not required on the finest grid level. Therefore, we operate on a coarser grid level l with respect to the storing of streamlines. This also reduces the number of required ghost boxes significantly. For instance we operate on level $l = 21$, where $k = 27$ in case of Chua's circuit and the Lorenz system.

5 A Further Application

In order to show how the presented techniques perform in a real-world application we apply them to a problem which arises in the design of trajectories for spacecrafts.

In 2001 the *Jet Propulsion Laboratory* plans to eject a craft into space which is supposed to stay on a periodic orbit between sun and earth for two years in order to collect solar matter (mission *Genesis*). One of the many challenges connected to this mission is to find the optimal trajectories for the transit between earth and the *Halo orbit* (and back). The central idea of the current approach is to look for them on the (*un*)stable manifold of the periodic orbit, see Figure 14. It is therefore indispensable to compute and visualize reliable approximations to these objects.

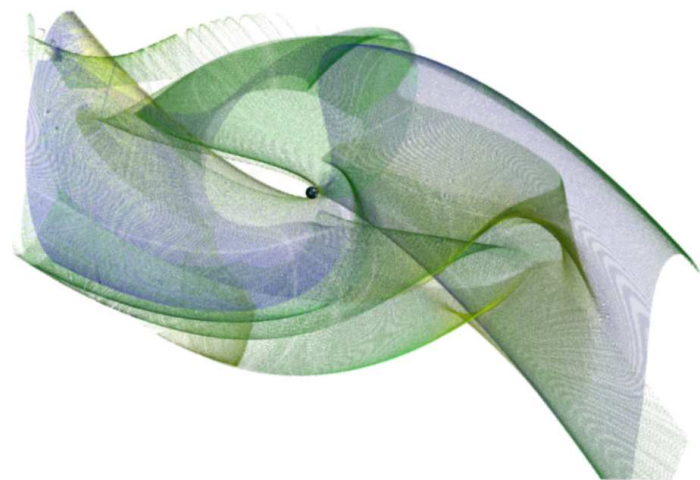


Figure 14: The unstable manifold of the periodic orbits which contains the transit paths of the spacecraft from earth to the Halo orbit and back. We have clipped the complete object at a suitable intersection plane. Color indicates the invariant measure, where yellow represents high probability measure and blue small measure values.

The mathematical model of the system {sun, earth/moon, spacecraft} is (in the final form) a six-dimensional ordinary differential equation of type (1) which is referred to as the *Circular Restricted Three Body Problem* – see e.g. [26] for an introductory exposition. The system possesses five equilibria. In the vicinity of one of them there exists a family of periodic (so called *Halo*-) orbits, one of which has been chosen as orbit of the spacecraft.

We use an adapted version of the globalization technique described in Section 2 to cover part of the unstable manifold of this Halo orbit. One might think of this object as the set of all possible ways back to earth. Figure 15 shows a dense coverage of part of the manifold with transit trajectories.

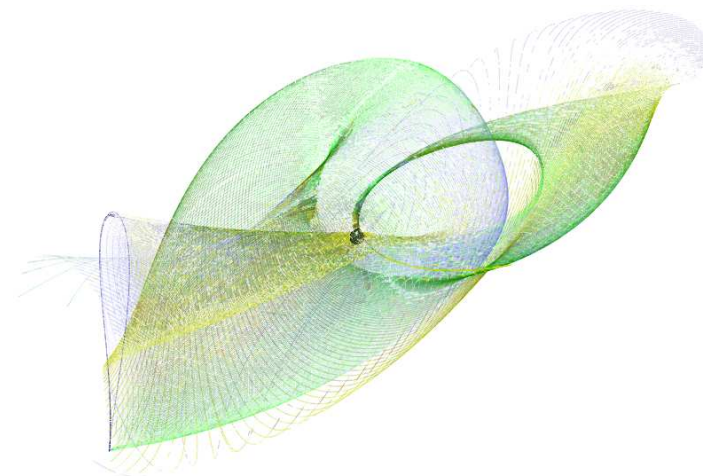


Figure 15: A part of the unstable manifold of the Halo-orbit, densely covered with possible trajectory paths of the spacecraft starting on the orbit. Color again indicates the invariant measure.

6 Conclusions and Future Work

We have presented a new approach to compute and efficiently visualize reliable approximations of invariant sets, which characterize dynamical systems topologically. The method is based on a hierarchy of successively refined box coverings. Fundamental probability measures can be approximated on the box hierarchy. We have explained how to make use of this approach to perceive dynamical systems in a novel, considerably expanded way. It provides an easy and intuitively understandable visual impressions of complex dynamical phenomena. Furthermore, we are able to combine the smoothly shaded representation of the invariant manifold's geometry with a coloring according to a probability measure and a dense coverage by streamlines, which also enlightens the local dynamics. Interesting future research directions are

- the incorporation of adaptive multiresolutional algorithms [21] in the computation as well as the visualization,
- if possible, a triangular surface reconstruction from the box covering making use of the approximate normal calculus,
- the appropriate animation of the long time behaviour of the dynamical system.

References

- [1] D. C. Banks. Illumination in Diverse Codimensions. *Computer Graphics Annual Conference Series*, pages 327–334, 1994.

- [2] B. Cabral and L. Leedom. Imaging vector fields using line integral convolution. In J. T. Kajiya, editor, *Computer Graphics (SIGGRAPH '93 Proceedings)*, volume 27, pages 263–272, Aug. 1993.
- [3] W. L. F. Degen and V. Milbrandt. The geometric meaning of nielson’s affine invariant norm. *CAGD*, 15:19–25, 1997.
- [4] M. Dellnitz, A. Hohmann, O. Junge, and M. Rumpf. Exploring invariant sets and invariant measures. *CHAOS: An Interdisciplinary Journal of Nonlinear Science*, 7(2):221, 1997.
- [5] M. Dellnitz and A. Hohmann. The computation of unstable manifolds using subdivision and continuation, in *Nonlinear Dynamical Systems and Chaos* (H.W. Broer, S.A. van Gils, I. Hoveijn und F. Takens eds.), *PNLDE 19* (Birkhäuser, 1996), 449-459.
- [6] Dellnitz, M. and Hohmann, A., A subdivision algorithm for the computation of unstable manifolds and global attractors. *Numerische Mathematik 75*, 293-317, 1997.
- [7] Dellnitz, M. and Junge, O., On the approximation of complicated dynamical behaviour. To appear in *SIAM J. Numer. Anal.*, 1998.
- [8] Eckmann, J.P. and Ruelle, D., Ergodic theory of chaos and strange attractors, *Rev. Mod. Phys.* **57**, 617-656, 1985.
- [9] G. Elber. Line Art Illustrations of Parametric and Implicit Forms. *IEEE Transactions on Visualization and Computer Graphics*, 4 (1):71–81, 1998.
- [10] I. Gargantini. Linear octrees for fast processing of three-dimensional objects. *Comput. Graph. Image Process.*, 20(4):365–374, 1982.
- [11] M. H. Ghavamnia and X. D. Yang. Direct rendering of laplacian compressed volume data. In *Proceedings of the Visualization*, pages 192–199, 1995.
- [12] Guckenheimer, J. and Holmes, P., *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Springer-Verlag, New York, 1983.
- [13] Hsu, H., Global analysis by cell mapping. *Int. J. Bif. Chaos* **2**, 727-771, 1992.
- [14] Huang, A., Pivka, L., Wu, C.W. and Franz, M., Chua’s equation with cubic nonlinearity, *Int. J. Bif. Chaos* **6**, 1996.
- [15] Kifer, Yu., Computations in dynamical systems via random perturbations. *Discrete and Continuous Dynamical Systems* **3**(4), 457-476, 1997.
- [16] D. Laur and P. Hanrahan. Hierarchical splatting: A progressive refinement algorithm for volume rendering. In T. W. Sederberg, editor, *Computer Graphics (SIGGRAPH '91 Proceedings)*, volume 25, pages 285–288, July 1991.
- [17] R. Lehoucq, D.C. Sorensen, and C. Yang. *ARPACK Users Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi methods*. SIAM Publications, Philadelphia, 1998.
- [18] N. Max. Optical Models for Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1 (2):99–108, 1995.
- [19] N. Max, R. Crawfis, and C. Grant. Visualizing 3D Velocity Fields Near Contour Surface. In *Proceedings of IEEE Visualization '94*, pages 248–254, 1994.
- [20] Metscher, M. and Rumpf, M. Optimal searching on hierarchical grids based on local coordinates. In *to appear in Proceedings of the GAMM- Seminar, Kiel*, 1998.
- [21] M. Ohlberger and M. Rumpf. Adaptive projection operators in multiresolutional scientific visualization. *IEEE Transactions on Visualization and Computer Graphics*, 4 (4), 1998.
- [22] P. Shirley and A. Tuchman. A Polygonal Approximation to Direct Scalar Volume Rendering. *Computer Graphics*, 24 (5):63–70, 1990.
- [23] D. Stalling, M. Zöckler, and H.-C. Hege. Fast display of illuminated field lines. *IEEE Transactions on Visualization and Computer Graphics*, 3(2), Apr.–June 1997. ISSN 1077-2626.
- [24] D.C. Sorensen. Implicit application of polynomial filters in a k-step arnoldi method. *SIAM J. Matr. Anal. Apps.*, 13:357–385, 1992.
- [25] M. Tamminen and H. Samet. Efficient octree conversion by connectivity labeling. *Computer Graphics*, 18(3):43–51, 1984.
- [26] Thurman, R. and Worfolk, P.A., The geometry of halo orbits in the circular restricted three body problem. *Geometry Center Research Report GCG95*, University of Minnesota, 1996.
- [27] J. J. van Wijk. Spot noise-texture synthesis for data visualization. In T. W. Sederberg, editor, *Computer Graphics (SIGGRAPH '91 Proceedings)*, volume 25, pages 309–318, July 1991.
- [28] J. J. van Wijk. Flow visualization with surface particles. *IEEE Computer Graphics and Applications*, 13(4):18–24, July 1993.
- [29] R. Wegenkittl, H. Löffelmann, and E. Gröller. Visualizing the Behavior of Higher Dimensional Dynamical Systems. In *Proceedings of IEEE Visualization '97*, pages 119–125, 1997.