The Orthogonal Constraints Problem with the Constraint Approach to Proxy-based Volume Haptics and a Solution

Karljohan Lundin* Matthew Cooper Anders Ynnerman

Norrköping Visualization and Interaction Studio
Linköping University, Sweden

Abstract

Recently the constraint approach to proxy-based volume haptics was introduced which provided a stable and effective means of conveying information about volumetric data through a haptic instrument. In this paper we present a proof that the approach is incapable of handling non-orthogonal constraints and discuss the implications of this restriction in detail. We also describe how full utilization of haptics applications in which multiple properties are used to enhance the understanding of complex data requires the use of non-orthogonal constraints. We then show how proxy-based volume haptics can be modified to allow for general constraints through the introduction of haptic primitives used to model the constraints. By balancing the forces exerted by the primitives on the proxy continuously, non-orthogonal constraints can be handled.

CR Categories: H.5.2 [Information Interfaces and Presentation]: User Interfaces; I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual reality

Keywords: Proxy-based volume haptics, orthogonal constraints, haptic primitives

1 Introduction

It has been shown that adding haptic feedback to an application can significantly increase both precision and speed of human-computer interaction. Scientific visualization and data exploration are no exceptions. Most research on haptics focuses on surfaces, so the lack of surfaces in volume data demands alternative methods that do not require explicit surface representations.

The first method available for interaction with volumetric data was the force functions-based approach[Iwata and Noma 1993; Mor et al. 1996; Avila and Sobierajski 1996; Hashimoto and Iwata 1997; Infed et al. 1999; Lawrence et al. 2000]. With this method the force is expressed as a function of the data represented. It is easy to implement and therefore quite popular, but suffers from instability. Also, representing all features as simple forces of varying strength and direction can, in some cases, be considered to be a too simplistic approach.

As an alternative, the proxy-based approach to volume haptics was introduced in [Lundin et al. 2002]. It is more suitable for representing shapes in the volumetric data. The method generates simple haptic constraints, described in detail in section 2, that yield to a certain force. Using these constraints surface feedback can be generated from scalar density data by using the gradient information in the data. Since the constraints, and thus also the surfaces, yield to a certain force this method avoids introducing haptic occlusion; in other words it does not physically obstruct the exploration. The method has also been generalized[Ikits et al. 2003] to generate other shapes than surfaces and also include vector and tensor fields.

The proxy-based approach complements the force function-based methods by introducing “passive” interaction, as opposed to the continuous force from a simple function. While a force function calculates the feedback solely from the features found in the local data, a proxy-based method has the ability to generate feedback in response to user actions, so that if the user applies no force, no force is fed back.

In this paper we present a proof that the proxy-based approach breaks down if the constraints become non-orthogonal. It is thus incapable of handling non-orthogonal constraints without introducing severe haptic artifacts. We also describe the need for support of non-orthogonal constraints in haptic interaction of volumetric data and what limitations the lack of such support force on a haptic environment.

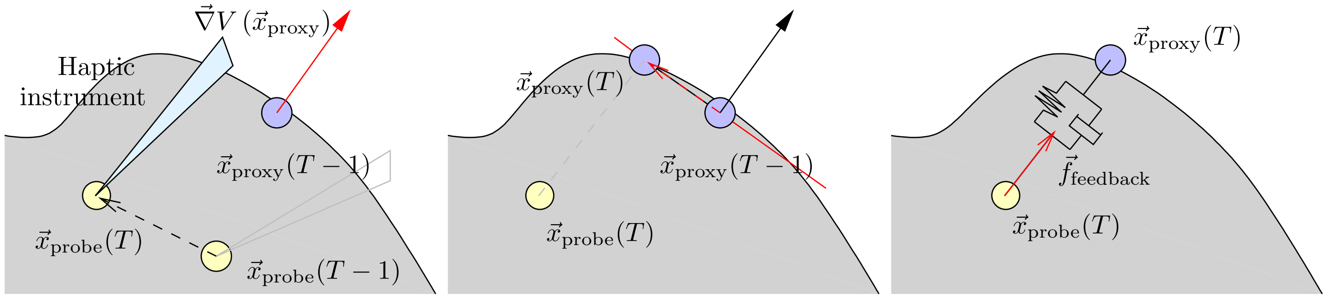
In [Lundin et al. 2005] we introduced haptic primitives as a way of modelling constraints from conceptually fundamental building blocks. By balancing the primitives it is also possible to include non-orthogonal constraints. In the second part of this paper we show how the use of haptic primitives, together with a numerical solver, enables the inclusion of general constraints in the haptic feedback loop, without the need for orthogonality.

2 Proxy-based Volume Haptics

In the proxy-based approach an internal proxy to the haptic probe is introduced. With the proxy, three simple steps are used in each time-frame of the haptic loop to produce the desired haptic effect from the constraints. First local data properties around the proxy point are determined. Then, depending on the local data, the proxy is moved a certain distance in local space. Finally the new proxy position is used to calculate the force feedback for the haptic instrument. These steps for proxy-based volume haptics are shown in figure 1.

1) Properties The continuous property fields needed to generate the haptic feedback are estimated from the discrete volumetric data through interpolation at the proxy position, see figure 1(a). For example the gradient vector field can be estimated from scalar data and curl or divergence may, if needed for generating the haptic effect, be extracted from

*e-mail: karlu@itn.liu.se



(a) Step 1: Evaluate local data properties. In this example the gradient vector, $\vec{\nabla} V(\vec{x}_{\text{proxy}})$, is extracted at the proxy position, $\vec{x}_{\text{proxy}}$.

(b) Step 2: Move the proxy point according to haptic constraints, in this case a to simulate the feeling of a plane.

(c) Step 3: Calculate feedback force by simulating spring-damper coupling.

Figure 1: Three steps for generating proxy-based haptic feedback, in this example from a virtual surface.

the vector data. These data are used in the second step to control the constraints that define the haptic effect. To produce more natural feedback from the volumetric data, transfer functions can be used to provide estimates of material properties directly from the data[Lundin et al. 2002; Avila and Sobierajski 1996; Aviles and Ranta 1999]. Even though using transfer functions for haptic feedback is not as established and extensively tested as using transfer functions for visual volume rendering, there is a close similarity between how they are used to define visual colours and to estimate material properties. Some examples of properties that are estimated and affect the haptic feedback are viscosity, friction, stiffness and flow strength.

2) Movements In the constraints approach simple one dimensional constraints, restricting the movement of the haptic instrument, are defined as functions of the properties of the local data. The strength of the constraint is controlled through a transfer function, as described above, and the direction of the constraint can be controlled by a vector property to produce a haptic representation of that property, for example the gradient. Each constraint controls the movement of the proxy in a direction to simulate a constraint in that direction. For a direction, represented by the unit vector $\hat{q}_i$, the equation that moves the proxy in the direction is formulated as

$$\vec{\eta} = \vec{x}_{\text{probe}} - \vec{x}_{\text{proxy}} \tag{1}$$

$$\vec{x}'_{\text{proxy}} = \vec{x}_{\text{proxy}} + \begin{cases} \hat{q}_i\left(\vec{\eta}\cdot\hat{q}_i - s_i/k\right), & \text{if } s_i < k\left(\vec{\eta}\cdot\hat{q}_i\right) \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $\vec{x}_{\text{proxy}}$ and $\vec{x}_{\text{probe}}$ are proxy and probe position, s_i is the strength of the constraint and k is the current stiffness used in the virtual coupling described below. By combining three independent orthogonal constraints in a local frame of reference a feeling of surfaces, friction, viscosity or transverse damping can be generated. The proxy movement is calculated separately in each direction and combined linearly to give the new proxy position, see figure 1(b).

3) Feedback After the new proxy position has been determined, the force feedback is calculated from a virtual spring-damper coupling the probe with the proxy point (see figure 1(c)). Thus the force feedback, $\vec{f}_{\text{feedback}}$, is evaluated through

$$\vec{f}_{\text{feedback}} = k\left(\vec{x}_{\text{proxy}} - \vec{x}_{\text{probe}}\right) + D\left(\vec{v}_{\text{proxy}} - \vec{v}_{\text{probe}}\right) \tag{3}$$

where $\vec{x}_{\text{proxy}}$ and $\vec{x}_{\text{probe}}$ are proxy and probe position, $\vec{v}_{\text{proxy}}$ and $\vec{v}_{\text{probe}}$ are proxy and probe velocity and k and D are stiffness and damper parameters, respectively.

3 Non-orthogonality Problem

As has already been mentioned, the above outlined constraint approach to proxy-based volume haptics is incapable of handling non-orthogonal constraints correctly. In this section we present a proof of this statement and discuss the impact and consequences of this.

3.1 Current Limitations

The constraint approach to proxy-based volume haptics will produce severe haptic artifacts when non-orthogonal features are encountered. This can be easily proven using a simple example with three constraints of zero strength. With no strength on the constraints (and a non zero stiffness, k) the proxy should end up at the probe position, so that the feedback from equation 3 yield zero.

Orthogonality requirement. Here it is shown that this requires orthogonality between the different constraints. From equation 2, by setting the strength, s_i, to zero and applying it three times, in different directions, we get

$$\vec{\eta} = \vec{x}_{\text{probe}} - \vec{x}_{\text{proxy}} \tag{4}$$

$$\vec{x}'_{\text{proxy}} = \vec{x}_{\text{proxy}} + \hat{q}_1\left(\vec{\eta}\cdot\hat{q}_1\right) + \hat{q}_2\left(\vec{\eta}\cdot\hat{q}_2\right) + \hat{q}_3\left(\vec{\eta}\cdot\hat{q}_3\right) \tag{5}$$

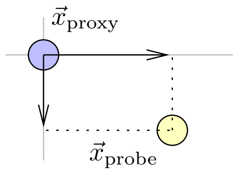
For the proxy, $\vec{x}_{\text{proxy}}$, to end up at the probe position, $\vec{x}_{\text{probe}}$, we see that

$$\begin{aligned} \vec{x}'_{\text{proxy}} &= \vec{x}_{\text{probe}} \\ &= \vec{x}_{\text{proxy}} + \vec{x}_{\text{probe}} - \vec{x}_{\text{proxy}} \\ &= \vec{x}_{\text{proxy}} + \vec{\eta} \end{aligned} \tag{6}$$

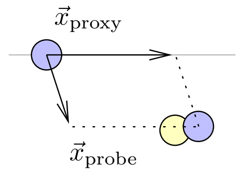
By combining equations 5 and 6 we get

$$\vec{\eta} = \hat{q}_1\left(\vec{\eta}\cdot\hat{q}_1\right) + \hat{q}_2\left(\vec{\eta}\cdot\hat{q}_2\right) + \hat{q}_3\left(\vec{\eta}\cdot\hat{q}_3\right) \tag{7}$$

which can only be true if the unit vectors, $\hat{q}$, are orthogonal. Thus, to get zero feedback from constraints with zero



(a) Orthogonal constraints move the proxy to the probe position.



(b) Non-orthogonal constraints move the proxy to an incorrect position.

Figure 2: Proxy movements with two constraints using the constraint-based approach. Both constraints have zero strength in this example and the proxy should be moved to the probe position.

strength, each constraint has to be orthogonal to the other constraints. □

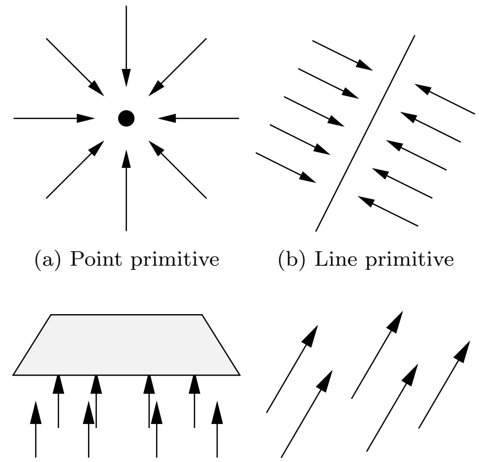
The individual movements due to non-orthogonal constraints will contribute to each other, which results in a different movement than desired. This effect is shown in figure 2. All combinations of non-orthogonal constraints show this behaviour and there is no simple way to circumvent this problem.

3.2 Impact

In research on and applications of haptics in scientific visualization, single properties have often been used to generate feedback from a single dataset at a time. Interacting with a single object at a time is common in visualization and exploration applications, however, as we move closer to immersive virtual reality environments this might change. In virtual reality (VR) applications it is common that the user interacts with multiple objects at a time. This has not been identified as a problem, since multiple surface objects can be easily handled by a user and it is the inter-object collision handling that has been the most challenging problem. With the constraint approach to volume haptics, however, interaction with volumetric data is restricted to single objects at a time. Thus as volume visualization and haptic force feedback are combined in virtual environments, the constraint approach will not suffice.

The biggest problem is, however, in the interaction with single objects of more advanced nature. Scientific visualization of multi-modal data is a growing area in which multiple datasets of the same object but of different modalities are co-located and co-registered to provide more information about the object than is possible with a single modality. An example is Computational Fluid Dynamics that produces both vector data describing the flow and several extra scalar datasets describing pressure, density, temperature, etc. The features of the different modalities are not guaranteed to be mutually orthogonal. Thus, the requirement of orthogonal constraints in the haptic interaction makes feedback from multi-modal data impossible.

Also in interaction and exploration of a single dataset there may be multiple properties that can be used to produce simultaneous information feedback. One example of this is the virtual wind-tunnel, where several properties of the vector data alone are interesting, such as the path of the flow, the strength of the flow and the vorticity. Since the constraint approach is incapable of handling non-orthogonal constraints simultaneously, the user must select only a single property to provide both information about the data and guide the user through the exploration process.



(a) Point primitive (b) Line primitive

(c) Plane primitive (d) Directed force

Figure 3: Forces from the haptic primitives.

4 Haptic Primitives

So far we have proved the existence of the orthogonality-problem and discussed its impact. In this section we describe our solution to the problem.

Using the single-dimensional constraints, more advanced constraints and types of behaviour are generated by rotating the constraints with respect to the haptic instrument. Our first step in the process of removing the orthogonality-problem is, instead, to provide one type of constraint for each desired type of haptic behaviour. The most basic way to discriminate types of constraints is by their dimensionality — constraints of one, two and three degrees of freedom. We call these *plane*, *line* and *point*, respectively, from the shape of their respective domain of constraint free motion. By adding a fourth type of haptic effect, *force*, we have a set of basic components that can be used to build any of the previously encountered haptic effects. We call these base components *haptic primitives*[Lundin et al. 2005].

The primitives provide a high level of abstraction that can be used to model a wide range of haptic modes and combinations of such, representing data from different scientific disciplines. They are designed to pull and push the haptic probe in well-defined directions to simulate features in the volume, see figure 3. As we will show this is done by controlling the proxy point movements. The haptic primitives and their properties are thus used to generate haptic feedback in a manner similar to that of how the constraints are used.

The haptic primitives can be expressed as simple force functions as shown below. It should be noted that the forces originating from the primitives act only on the proxy and are a means to find the new proxy position. This is done by balancing the net force from the primitives against the force feedback from equation 3 (see example in figure 4). This is presented in detail in section 4.2. After the new proxy position is found, the force feedback is calculated through the virtual coupling, equation 3.

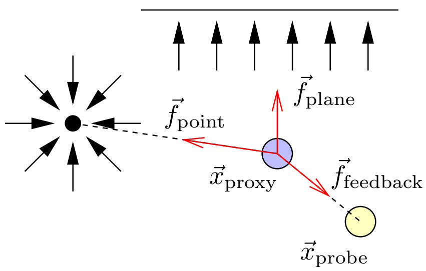


Figure 4: Finding the proxy position that balances the force between plane and point primitive ($\vec{f}_{\text{plane}}$ and $\vec{f}_{\text{point}}$, respectively) and feedback force ($\vec{f}_{\text{feedback}}$).

4.1 Intermediate Force Representation

In this section we describe the haptic primitives and how they, from their position and strength, define the force field that affects the proxy. For primitive i at position $\vec{x}_i$ having a strength of s_i, if the primitive has a direction/orientation then this is defined by a unit vector, $\hat{q}_i$. The proxy position is, as before, denoted $\vec{x}_{\text{proxy}}$.

The directed force is the simplest primitive. It has no position and generates a force $\vec{f}_i$ defined by

$$\vec{f}_i\left(\vec{x}_{\text{proxy}}\right) = s_i \hat{q}_i \tag{8}$$

This primitive can be used to simulate gravity or magnetic attraction. It also provides a means for integrating force-function feedback with the proxy-based approach.

The point primitive attracts the proxy towards the position of the primitive. With the displacement of the proxy relative to the primitive being $\vec{x}_i - \vec{x}_{\text{proxy}}$, we calculate the force by

$$\vec{f}_i\left(\vec{x}_{\text{proxy}}\right) = \begin{cases} \vec{0}, & \text{if } |\vec{x}_i - \vec{x}_{\text{proxy}}| = 0 \\ s_i \frac{\vec{x}_i - \vec{x}_{\text{proxy}}}{|\vec{x}_i - \vec{x}_{\text{proxy}}|}, & \text{if } |\vec{x}_i - \vec{x}_{\text{proxy}}| \neq 0 \end{cases} \tag{9}$$

The uniform feedback from this primitive makes it suitable for such effects as viscosity.

The line primitive has both position, strength and orientation. It attracts the proxy towards the closest point on the line defined by the position and the direction vector $\hat{q}_i$. With a vector $\vec{m}$, pointing from the proxy to the closest point on the line, being expressed by

$$\vec{m} = \hat{q}_i\left[\hat{q}_i \cdot \left(\vec{x}_{\text{proxy}} - \vec{x}_i\right)\right] - \left(\vec{x}_{\text{proxy}} - \vec{x}_i\right) \tag{10}$$

we calculate the force by

$$\vec{f}_i\left(\vec{x}_{\text{proxy}}\right) = \begin{cases} \vec{0}, & \text{if } |\vec{m}| = 0 \\ s_i \frac{\vec{m}}{|\vec{m}|}, & \text{if } |\vec{m}| \neq 0 \end{cases} \tag{11}$$

The plane primitive is most similar to the simple constraint in that it produces a yielding restraining force in one single direction. To generate this effect it attracts the proxy in the direction of the surface normal, $\hat{q}_i$, but only when the proxy is on the negative side of the surface, i.e. when

$$\left(\vec{x}_{\text{proxy}} - \vec{x}_i\right) \cdot \hat{q}_i < 0 \tag{12}$$

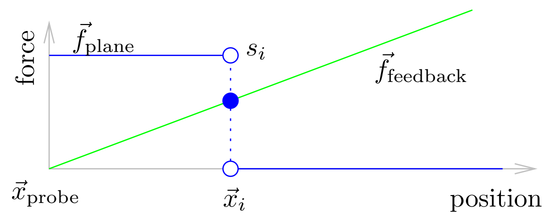


Figure 5: Finding the static equilibrium with a plane primitive — balance between the force feedback from equation 3 and the plane primitive at position x_i, equation 13.

As long as equation 12 holds, the force from the plane primitive is constant, so we define the force by

$$\vec{f}_i\left(\vec{x}_{\text{proxy}}\right) = \begin{cases} 0, & \text{if } \left(\vec{x}_{\text{proxy}} - \vec{x}_i\right) \cdot \hat{q}_i \geq 0 \\ s_i \hat{q}_i, & \text{if } \left(\vec{x}_{\text{proxy}} - \vec{x}_i\right) \cdot \hat{q}_i < 0 \end{cases} \tag{13}$$

4.2 Finding the new Proxy Position

To simulate the haptic feedback, the proxy is moved to a new position in every time-frame. This new position is found by balancing the force feedback from equation 3 with the force from the haptic primitives involved, as is shown in figure 4. In our work we disregard the damping term to simplify the calculations, that is we set $D = 0$ in equation 3. Suppose that we have sets of directed force, point, line and plane primitives denoted $\mathcal{A}_{\text{directed}}$, $\mathcal{A}_{\text{point}}$, $\mathcal{A}_{\text{line}}$ and $\mathcal{A}_{\text{plane}}$, respectively. We calculate the residual force, $\vec{f}_{\text{res}}$, from the primitives and the force feedback by

$$\begin{aligned} \vec{f}_{\text{res}} &= -k\left(\vec{x}_{\text{proxy}} - \vec{x}_{\text{probe}}\right) \\ &+ \sum_{i \in \mathcal{A}_{\text{directed}}} s_i\, \hat{q}_i \\ &+ \sum_{i \in \mathcal{A}_{\text{point}}} \begin{cases} 0, & \text{if } |\vec{x}_i - \vec{x}_{\text{proxy}}| = 0 \\ s_i \frac{\vec{x}_i - \vec{x}_{\text{proxy}}}{|\vec{x}_i - \vec{x}_{\text{proxy}}|}, & \text{if } |\vec{x}_i - \vec{x}_{\text{proxy}}| \neq 0 \end{cases} \\ &+ \sum_{i \in \mathcal{A}_{\text{line}}} \begin{cases} 0, & \text{if } |\vec{m}| = 0 \\ s_i \frac{\vec{m}}{|\vec{m}|}, & \text{if } |\vec{m}| \neq 0 \end{cases} \\ &+ \sum_{i \in \mathcal{A}_{\text{plane}}} \begin{cases} 0, & \text{if } \left(\vec{x}_{\text{proxy}} - \vec{x}_i\right) \cdot \hat{q}_i \geq 0 \\ s_i \hat{q}_i, & \text{if } \left(\vec{x}_{\text{proxy}} - \vec{x}_i\right) \cdot \hat{q}_i < 0 \end{cases} \end{aligned} \tag{14}$$

The new proxy position is then found by solving

$$\vec{f}_{\text{res}}\left(\vec{x}_{\text{proxy}}\right) = \vec{0} \tag{15}$$

Equation 15 is solved using a numerical solver that searches for a best match by minimizing the magnitude of the function. The minimum magnitude of the residual force will yield the same proxy position as an analytical solution would (see figure 5).

As an example of the effects produced by our approach consider only the plane primitive in balance with the force feedback: the proxy will be positioned on the plane as long as the force feedback projected on the plane normal is less than or equal to the strength of the plane primitive. Then the setup behaves just like surface haptics, see figure 6, steps a to c. When the probe is moved further away from the plane, the proxy will find an equilibrium below the plane, giving the same effect as the constraints of earlier methods, see figure 6, step d. A similar effect is also generated by the point and the line primitive.

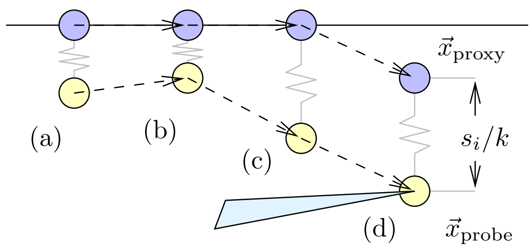


Figure 6: How the proxy is moved over a plane primitive when the probe is moved. The maximum distance between the proxy and probe positions induced by the haptic primitive is given by the primitive strength divided by the stiffness in equation 3, i.e. s_i/k.

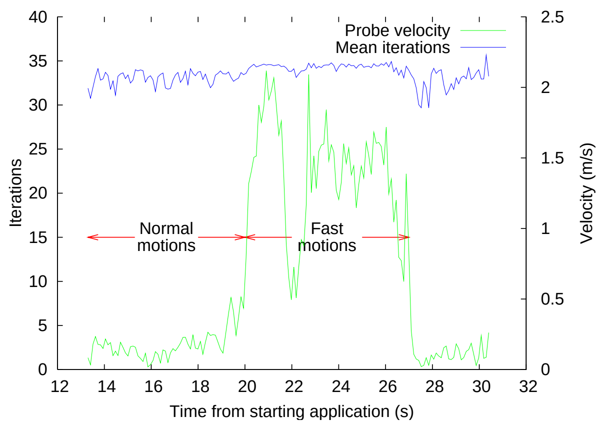


Figure 7: Number of iterations needed for our numerical solver to converge during a haptic exploration. The velocity of the probe is also presented.

4.3 The Numerical Solver

The solver we use to find the best match is a simple Euler solver. The problem is a non-linear minimization problem that can be solved using alternative iterative methods, such as the Nelder-Mead Solver. We have, however, so far found no analytical solution.

The precision and performance of the numerical solver is of utmost importance since it is the part of the algorithm that determines the proxy position for each time-frame and is the most computationally intensive part of the system. To speed up the convergence for the solver while maintaining high precision for the result we deploy an adaptive step-length approach. For two haptic primitives we find the solution in, typically, 30–40 iterations which takes less than 100 μs on our current hardware. The time complexity of the solver with respect to the number of haptic primitives is $\mathcal{O}(N)$. A graph describing the solver behaviour over a haptic exploration can be found in figure 7.

5 Conclusions

We have shown that the current constraint-oriented proxy-based approach to volume haptics has a severe limitation: all features represented by the haptic feedback *must* be orthogonal for the algorithm to work. A wide adoption of haptics for scientific visualization needs a solution to this problem. With this limitation haptic feedback can not be generated from an interaction with multiple objects or with multi-modal data.

To solve the problem, all constraints must be handled concurrently. This is done in our new primitive-based algorithm for volume haptics, introduced in [Lundin et al. 2005]. The method uses the notion of haptic primitives both as a comprehensive abstraction layer for implementing haptic schemes and as an effective means of calculating the haptic force feedback. Our solution to the orthogonal constraints problem allows for simultaneous feedback from multiple objects, haptic interaction with multi-modal data and simultaneous haptic interaction with multiple properties from a single dataset.

References

Avila, R. S., and Sobierajski, L. M. 1996. A haptic interaction method for volume visualization. In *Proceedings at IEEE Visualization*, 197–204.

Aviles, W., and Ranta, J. 1999. Haptic interaction with geoscientific data. In *Proceedings at Phantom User Group Workshop'99*.

Hashimoto, W., and Iwata, H. 1997. A versatile software platform for visual/haptic environment. *Journal of Control, Automation and Systems Engineering*, 106–114.

Ikits, M., Brederson, J. D., Hansen, C. D., and Johnson, C. R. 2003. A constraint-based technique for haptic volume exploration. In *Proceedings of IEEE Visualization '03, pp. 263–269*.

Infed, F., Brown, S. V., Lee, C. D., Lawrence, D. A., Dougherty, A. M., and Pao, L. Y. 1999. Combined visual/haptic rendering modes for scientific visualization. In *Proceedings of 8th Annual Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*.

Iwata, H., and Noma, H. 1993. Volume haptization. In *Proceedings of IEEE 1993 Symposium on Research Frontiers in Virtual Reality*, 16–23.

Lawrence, D. A., Lee, C. D., Pao, L. Y., and Novoselov, R. Y. 2000. Shock and vortex visualization using a combined visual/haptic interface. In *Proceedings of IEEE Conference on Visualization and Computer Graphics*.

Lundin, K., Ynnerman, A., and Gudmundsson, B. 2002. Proxy-based haptic feedback from volumetric density data. In *Proceedings of Eurohaptics*, University of Edinburgh, United Kingdom, 104–109.

Lundin, K., Gudmundsson, B., and Ynnerman, A. 2005. General proxy-based haptics for volume visualization. In *Proceedings of the World Haptics Conference*, IEEE, 557–560.

Mor, A., Gibson, S., and Samosky, J. 1996. Interacting with 3-dimensional medical data: Haptic feedback for surgical simulation. In *Proceedings of Phantom User Group Workshop'96*.