Eurographics Conference on Visualization (EuroVis) (2015)
E. Bertini, J. Kennedy and E. Puppo (Editors)



# Interaction with uncertainty in visualisations

Advait Sarkar [†1] Alan F Blackwell[1] Mateja Jamnik[1] Martin Spott[2]

[1]Computer Laboratory, University of Cambridge, United Kingdom
[2]BT Research & Technology, Adastral Park, Ipswich, United Kingdom

**Abstract**

*In recent years, a number of algorithms have been developed which provide fast approximate computations on large datasets. There is considerable interest in exploiting these techniques to render interactive visualisations of large datasets. Such interactions require careful consideration of the uncertainty arising from the approximations that these algorithms employ. Characterising and comparing visualisations of uncertainty has been well studied; however, it is typically assumed that uncertainty is a fixed property of the data. In light of this new generation of approximate visualisations, we consider the case where uncertainty arises from algorithms whose parameters can be altered, and so can be manipulated just like any other aspect of the visualisation. We present a novel direct-manipulation interface for uncertainty in visualisations and show through a user study that our interface enables people to successfully edit and comprehend uncertainty.*

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [Information Interfaces and Presentation]: User Interfaces—Interaction Styles

## 1 Introduction

The challenge of interaction with large datasets is twofold: it is a technical challenge to build hardware and software capable of processing them, but it is also an interaction design challenge to give users an understanding of and a sense of control over analytical processes; to create a perception of "agency" [CMK*12].

Techniques for *approximate computation* such as sampling [CDN07], sketching [CG07], and online aggregation [HHW97] appear to be a promising technical solution. These allow for fast processing of large datasets in exchange for quantifiable error bounds. Such advances have led to the development of database tools that can perform fast approximate queries [AMP*13]. There is considerable interest in using these techniques to improve the quality of interaction with large datasets, which is contingent on the visualisation being rendered at interactive speeds [KBP*14, SBJS14a].

The technical solution of approximate computation has a corresponding design problem: interaction with parameterised uncertainty. There are several ways of visualising uncertainty [THM*05, ZC07], including error bars, use of colour, transparency, blurring, painterly rendering, and animation. How these are categorised and evaluated is the subject of much research [ZC06, BBIF12]. However, it is typically assumed that uncertainty is a fixed aspect of the data, resulting either from the way in which the data was generated, or introduced by some fixed transformation of the raw data that has resulted in the current dataset. With the emergence of approximate computation techniques, we now have sources of uncertainty that make a trade-off between accuracy and time/space resources.

Consequently, there is a need for interfaces where parameterised uncertainty can be interacted with and manipulated [FPD*12, MPG*14], just as much previous attention has been paid to visual interaction with other types of statistical techniques [EHM*11, SHB*14, BLBC12, SBJS14b]. In this paper, we present the first visual direct manipulation interface designed to facilitate such interaction, and a user study showing that our interface is fit for this purpose.

## 2 The draggable error bars interface

Our interface is shown in Fig. 1. It is written in JavaScript, HTML and CSS, using the `canvas` element. We focus on scatterplots here, but our interface can be directly applied to other common chart types, such as bar and line graphs.

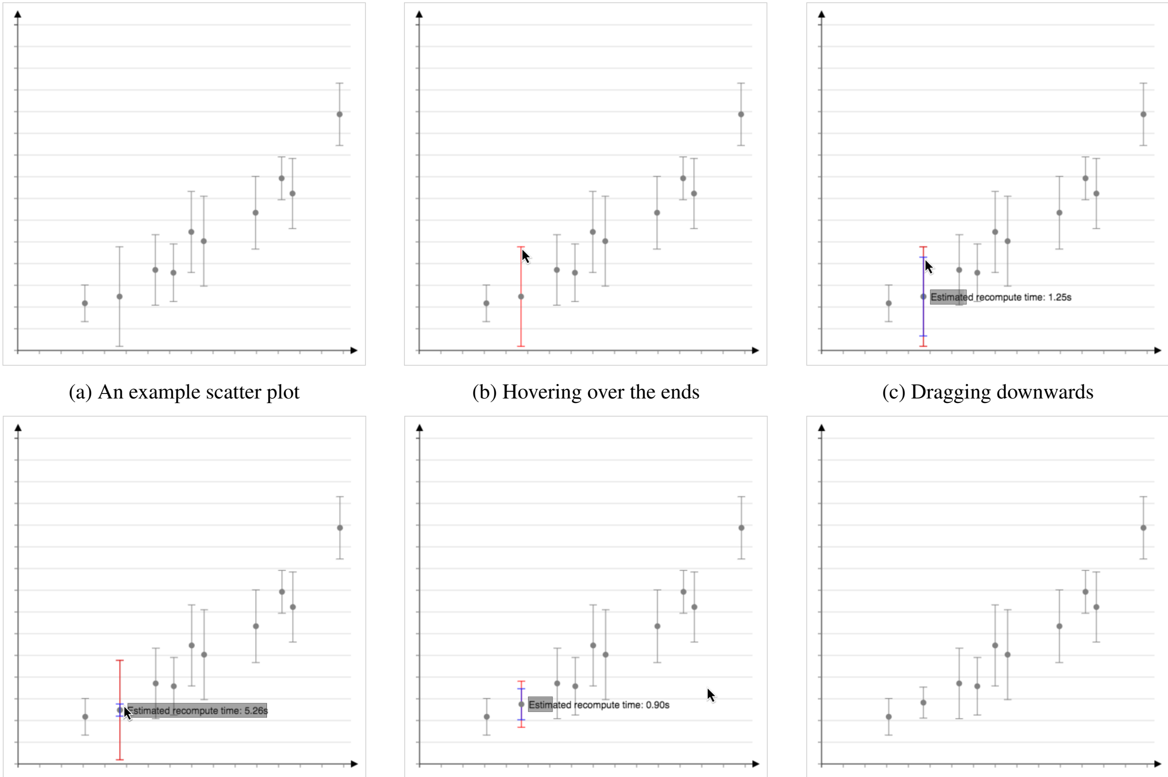
The uncertainty in the $y$-coordinates of the points due to approximate computation is represented using error bars. We use error bars in preference to other methods such as blurring and transparency for two reasons: first, because they provide clear drag targets (which blurring and transparency do not), and second, because varying degrees of uncertainty are more easily comparable using the lengths of error bars as opposed to opacity or blur radius [CM84] (although they are known to have some weaknesses [CG14]).

The user drags the ends of these error bars in order to reduce the uncertainty associated with those points. When dragging, a horizontal indicator bar appears, visualising the

† Advait is supported by an EPSRC iCASE award sponsored by BT, and also by a Computer Laboratory premium studentship.



DOI: 10.2312/eurovisshort.20151138




(a) An example scatter plot (b) Hovering over the ends (c) Dragging downwards

(d) Dragging further (e) Recomputation in progress (f) Recomputation complete

Figure 1: A sequence of images showing how the interface can be used to reduce the $y$-uncertainty associated with a point.

estimated duration of recomputation. This "resource cost estimation bar" is an essential component of the interface, as it allows the user to judge whether they are willing to invest their resources (in this case, time) in exchange for an improvement in accuracy.

The user can request a specific amount of uncertainty by dragging the bars varying amounts. Upon reaching the desired uncertainty, the user stops dragging (releases the mouse button), which starts the recomputation. While the recomputation is being performed, the cost estimation indicator shrinks, and the point and error bars move to their newly accurate positions.

### 2.1 Applicability of the draggable bars interface

For this interface to be applicable to an approximate computation technique, the technique must fulfil the following:

1. It must be possible to control some measure of the accuracy of the computation. This is necessary to provide semantics for the error bars.
2. It must be possible to estimate the usage of some resource (typically time or memory) which is required to carry out the computation. This is necessary to provide semantics for the cost estimation bar.

Examples of techniques that fulfil these criteria include computation driven by Bloom filters [Blo70], where the error bars can be mapped to false positive rates, and the cost estimation bar can be mapped to the length of the bit array. Similarly, in count-min sketches [CM05], error bars and cost estimation indicator(s) can be mapped to collision rates and bit array parameters respectively. In online aggregation, error bars and estimation bars can be mapped respectively to confidence intervals and query execution time. Thus, draggable error bars lend themselves to be a general direct-manipulation interface for controlling the parameters in several common approximate computation techniques.

We make a distinction between "big data" and "big visualisations". Our tool directly facilitates analysis of the former but not the latter. That is, our interface is most suitable for visualisations with a large underlying dataset, but nonetheless a small number of graphical elements. For instance, a database of 10 billion integers can be used to produce a plot of 10 points, where each point is meant to be the average of 1 billion integers. This is big data, but a small visualisation.

In our prototype, the errors as well as recomputations are simulated, remaining independent of any particular approximate technique. Hidden from the user, the simulation initialises each point with a true random $y$-value $y$, and random $y$-error $e$. For the user, a simulated approximate point $\hat{y}$ is rendered such that $\hat{y} \sim U(y+e, y-e)$. Upon dragging, the required $y$-error, $e'$, is set by the new length of the error bar. The estimated recompute time is calculated as an increasing function of $1-(e'/e)$. A new value for $\hat{y}$ is drawn from $U(y+e', y-e')$. Thus, $\lim_{e' \to 0} \hat{y} = y$. Although far more sophisticated treatments of error bar semantics are possible [Cum12], this simple generic simulation is a first step towards effectively representing interaction with the kinds of approximate techniques previously outlined.

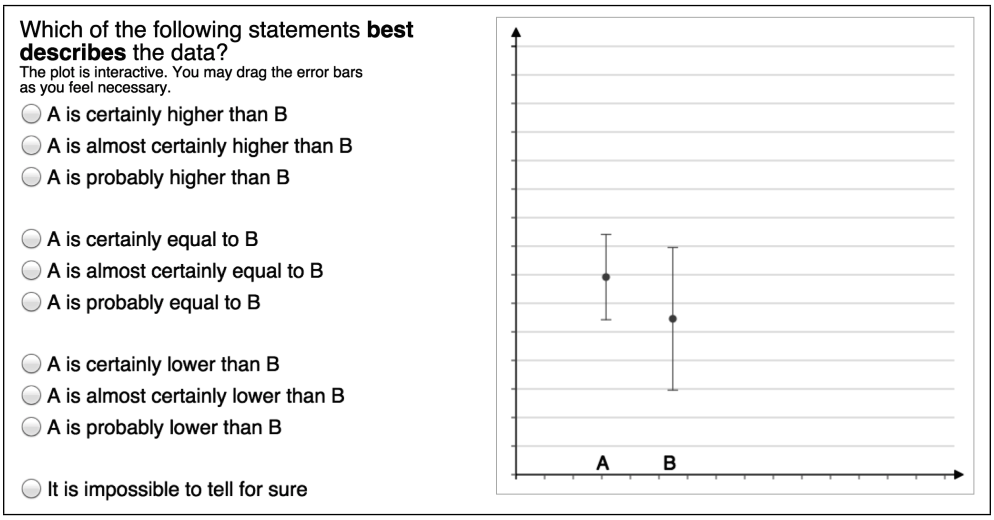



Figure 2: An example randomly-generated binary comparison task. Here, the participant may reasonably pick 'A is probably higher than B' or 'It is impossible to tell for sure', but they may also choose to refine the error bars before answering.

## 3 User study

We conducted a web-based study of 39 participants, recruited largely from English-speaking countries using the Microworkers platform (https://microworkers.com), a service similar to Amazon Mechanical Turk.

We were primarily interested in the following: **1**. Can users spontaneously identify our interface as enabling control over uncertainty? **2**. Do they use the interface? **3**. Do they use the interface correctly? **4**. Is their usage affected by the speed of the recomputation?

### 3.1 Spontaneous discovery of interface mechanism

Participants were first shown a plot with error bars, and asked to describe what they saw. 33 (85%) identified it as a plot or graph depicting data (as opposed to a picture or other type of image). However, only 8 (21%) stated that there was some uncertainty associated with the points on the plot.

After being given a brief explanation of error bars, participants were asked to reduce the uncertainty associated with a point on the plot with no further instructions. 34 participants (87%) successfully discovered the drag operation. The median discovery time for the drag operation was 19.3 seconds. From this we conclude that the interface corresponds well with users' prior assumptions about how one might manipulate uncertainty in a visualisation, i.e., it is *intuitive*.

### 3.2 Do people use the interface?

Next, participants were given full explanations of the interface and how to use it. Thereafter, each participant performed 30 comparison tasks in which they were shown a plot with two points and asked to choose the best description of the points out of a standard list of options. An example task is shown in Fig. 2.

It was always possible to give a reasonable interpretation of the plot without dragging the error bars. Even so, 30 participants (77%) used drag operations during the experiment. 13 participants used drag operations in all 30 tasks, averaging 2.53 drags per task (Fig. 3 shows the full distribution of drag counts per task), dragging each point an average of 1.26 times. On average, they chose to reduce error by 67.6% per drag. Fig. 4 shows the full distribution of uncertainty reduction per drag (this corresponds to $1-(e'/e)$ as in §2.1).

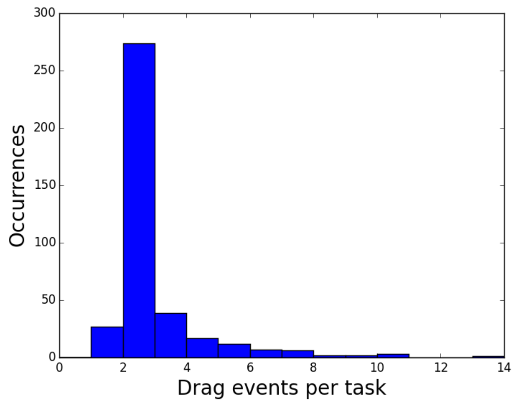



Figure 3: Distribution of drag counts per task.

### 3.3 Is the interface used correctly?

For any given plot configuration, we used the following rules to construct a set of 'reasonable' answers:

- Where the error bars are not overlapping, the 'certainly' and 'almost certainly' unequal options (i.e. higher/lower) are reasonable in the correct direction of the inequality.
- Where the error bars are overlapping:
  - The 'impossible to tell' option is reasonable.
  - If the top and bottom of point A's error bars are close (<0.5cm) to the top and bottom of B's error bars respectively, the 'probably equal' option is reasonable.
  - If 'probably equal' is reasonable and the vertical lengths of A's and B's error bars are small (<0.5cm), the 'almost certainly equal' option is reasonable.

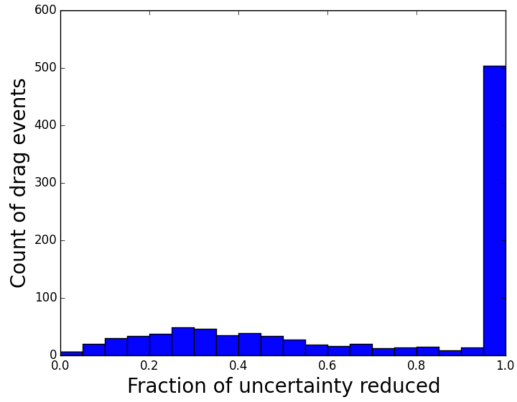


Figure 4: Uncertainty reduction per drag (0.5 bins).

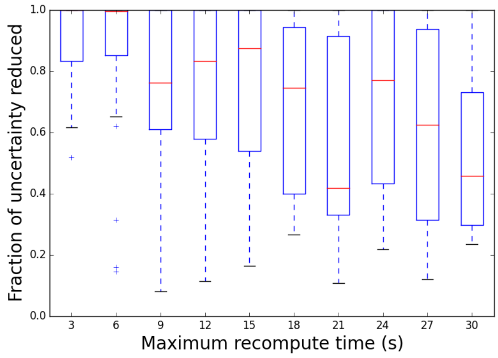


Figure 5: Boxplots of uncertainty reductions per drag per task, against maximum recompute time for that task. Observe the medians getting lower with increasing times.

  – The 'probably' and 'almost certainly' unequal options are reasonable, with the correct direction determined by comparing $P(A \geq B)$ and $P(A \leq B)$, where $A$ and $B$ are uniformly distributed between their respective error bounds.

- Where the error bars have been completely reduced, the 'certainly' and 'almost certainly' options are considered reasonable in the direction of the inequality, or if the points are equal, the 'equals' case is reasonable.

It follows that for any configuration of points and error bars, there is always a non-empty set of reasonable answers, whether or not the user has chosen to manipulate that plot.

In 89.5% of tasks where a reasonable answer was given *and* a drag operation was performed, the drag operation was actually necessary to be able to provide that answer. That is, the participant's answer was in the set of reasonable interpretations for the state of the plot *after* performing one or more drag operations, but *not* in the set of reasonable interpretations for the initial state of the plot. This often corresponded to users dragging overlapping error bars until they no longer overlapped, and then selecting a 'certainly' option. Thus, the participants applied the interface correctly to arrive at reasonable answers that were not deducible without using the interface.

### 3.4 Is usage affected by recomputation time?

Since we are interested in the effect of the speed of the recomputation on dragging behaviour, we now focus on the 13 participants who used drag operations in every task. The tasks were randomly generated to produce a maximum recompute time between 3 and 30 seconds inclusive, in 3-second intervals. There is a slight correlation between maximum recompute time and drag count per task ($r = 0.1, p = 0.046$), and a clear negative correlation between maximum recompute time and uncertainty reduced per drag per task ($r = -0.25, p = 5.65 \cdot 10^{-7}$). This corresponds to behaviour observed in our pilot studies where participants preferred to make multiple small drags if recomputation was expensive.

Moreover, in Fig. 4 we observe a clear tendency to reduce uncertainty completely (i.e. drag the error bars all the way), but an interesting distribution arises around the lower levels. This turns out to be attributable to a behavioural divide between participants. The 13 participants considered in this subsection can be clearly subdivided into two groups: those who tend to eliminate uncertainty entirely, and those who vary their behaviour from task to task. There were 4 participants in the first group, who each reduced uncertainty by over 99.9% per task (i.e., fully dragged the error bars in every task); this may simply be due to the fact that the recompute times were generally quite small. The remainder were in the latter, behaviour-varying group; on average members of this group reduced uncertainty by only 71% per task.

Upon examination of just the 9 participants who exhibit varying behaviour, the effect of maximum recompute time is even more apparent. While the relationship between maximum recompute time and drag count per task remains essentially unchanged ($r = 0.13, p = 0.034$), the correlation between maximum recompute time and uncertainty reduced per drag per task is much stronger in these participants ($r = -0.35, p = 3.56 \cdot 10^{-9}$). This is illustrated in Fig. 5.

From these observations, we conclude that usage is affected by recomputation speed in skilled users. This relates to Liu and Heer's conclusions that latency in visualisations modulates interaction [LH14]. In the context of our interface, this can be interpreted as the interface successfully communicating to the user the implications of their actions, allowing them to make informed decisions about computations on their data (similar to Fisher et al [FPD*12]).

## 4 Conclusion

We present a novel direct-manipulation interface for interacting with uncertainty arising from approximate computations. The interface takes the form of draggable error bars, which represent uncertainty, and a cost estimation indicator, which represents required computational resources (e.g. time or memory). The interface can be applied to many types of visualisation (e.g. scatter plots, bar charts, line graphs, etc.) and can be used to enable interaction with several types of approximate computation techniques (e.g. sampling, sketching, online aggregation, etc.).

We present results of a user study confirming that our system can be understood and applied successfully. Our study provides evidence that the representation of required computational resources modulates interaction as intended, allowing skilled users to modify their usage to achieve the resource/accuracy tradeoff that suits their needs.

## References


[AMP*13] AGARWAL S., MOZAFARI B., PANDA A., MILNER H., MADDEN S., STOICA I.: BlinkDB: queries with bounded errors and bounded response times on very large data. In *Proceedings of the 8th ACM European Conference on Computer Systems - EuroSys '13* (New York, New York, USA, 2013), ACM Press, p. 29. 1

[BBIF12] BOUKHELIFA N., BEZERIANOS A., ISENBERG T., FEKETE J.-D.: Evaluating Sketchiness as a Visual Variable for the Depiction of Qualitative Uncertainty. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (Dec. 2012), 2769–2778. 1

[BLBC12] BROWN E. T., LIU J., BRODLEY C. E., CHANG R.: Dis-function: Learning distance functions interactively. In *Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on* (2012), IEEE, pp. 83–92. 1

[Blo70] BLOOM B. H.: Space/time trade-offs in hash coding with allowable errors. *Commun. ACM 13*, 7 (July 1970), 422–426. 2

[CDN07] CHAUDHURI S., DAS G., NARASAYYA V.: Optimized stratified sampling for approximate query processing. *ACM Transactions on Database Systems (TODS) 32*, 2 (2007), 9. 1

[CG07] CORMODE G., GAROFALAKIS M.: Sketching probabilistic data streams. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data* (2007), ACM, pp. 281–292. 1

[CG14] CORRELL M., GLEICHER M.: Error bars considered harmful: Exploring alternate encodings for mean and error. *Visualization and Computer Graphics, IEEE Transactions on 20*, 12 (Dec 2014), 2142–2151. 1

[CM84] CLEVELAND W. S., MCGILL R.: Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American statistical association 79*, 387 (1984), 531–554. 1

[CM05] CORMODE G., MUTHUKRISHNAN S.: An improved data stream summary: the count-min sketch and its applications. *Journal of Algorithms 55*, 1 (2005), 58–75. 2

[CMK*12] COYLE D., MOORE J., KRISTENSSON P. O., FLETCHER P., BLACKWELL A.: I did that! measuring users' experience of agency in their own actions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 2025–2034. 1

[Cum12] CUMMING G.: *Understanding the new statistics: Effect sizes, confidence intervals, and meta-analysis*. Routledge, 2012. 2

[EHM*11] ENDERT A., HAN C., MAITI D., HOUSE L., LEMAN S., NORTH C.: Observation-level interaction with statistical models for visual analytics. In *Visual Analytics Science and Technology (VAST), 2011 IEEE Conference on* (Oct 2011), pp. 121–130. 1

[FPD*12] FISHER D., POPOV I., DRUCKER S., ET AL.: Trust me, i'm partially right: incremental visualization lets analysts explore large datasets faster. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2012), ACM, pp. 1673–1682. 1, 4

[HHW97] HELLERSTEIN J. M., HAAS P. J., WANG H. J.: Online aggregation. *ACM SIGMOD Record 26*, 2 (1997), 171–182. 1

[KBP*14] KIM A., BLAIS E., PARAMESWARAN A., INDYK P., MADDEN S., RUBINFELD R.: Rapid Sampling for Visualizations with Ordering Guarantees. 17. 1

[LH14] LIU Z., HEER J.: The Effects of Interactive Latency on Exploratory Visual Analysis. *IEEE Transactions on Visualization and Computer Graphics (Proceedings InfoVis 2014)*, August (2014). 4

[MPG*14] MUHLBACHER T., PIRINGER H., GRATZL S., SEDLMAIR M., STREIT M.: Opening the black box: Strategies for increased user involvement in existing algorithm implementations. *Visualization and Computer Graphics, IEEE Transactions on 20*, 12 (Dec 2014), 1643–1652. 1

[SBJS14a] SARKAR A., BLACKWELL A. F., JAMNIK M., SPOTT M.: Hunches and Sketches: rapid interactive exploration of large datasets through approximate visualisations. *The 8th International Conference on the Theory and Application of Diagrams, Graduate Symposium (Diagrams 2014)* (2014). 1

[SBJS14b] SARKAR A., BLACKWELL A. F., JAMNIK M., SPOTT M.: Teach and try: A simple interaction technique for exploratory data modelling by end users. In *Visual Languages and Human-Centric Computing (VL/HCC), 2014 IEEE Symposium on* (July 2014), IEEE, pp. 53–56. 1

[SHB*14] SEDLMAIR M., HEINZL C., BRUCKNER S., PIRINGER H., MOLLER T.: Visual parameter space analysis: A conceptual framework. *Visualization and Computer Graphics, IEEE Transactions on 20*, 12 (Dec 2014), 2161–2170. 1

[THM*05] THOMSON J., HETZLER E., MACEACHREN A., GAHEGAN M., PAVEL M.: A typology for visualizing uncertainty. In *Electronic Imaging 2005* (2005), International Society for Optics and Photonics, pp. 146–157. 1

[ZC06] ZUK T., CARPENDALE S.: Theoretical analysis of uncertainty visualizations. *Proc. SPIE 6060* (Jan. 2006), 606007–606014. 1

[ZC07] ZUK T., CARPENDALE S.: Visualization of Uncertainty and Reasoning. vol. 4569 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 164–177. 1