

# Sudoku Vis

## - How to Explore Relationships of Mutual Exclusion -

Gudrun Klinker\*

Technische Universität München, Fakultät für Informatik, Boltzmannstraße 3, 85748 Garching bei München, Germany

**Abstract.** In recent years, many systems for visualizing and exploring massive amounts of data have emerged. A topic that has not yet been investigated much concerns the visualization and exploration of constraints between objects such as relationships of joint contribution to a set or mutual exclusion. In this paper, we present concepts toward visualizing mutual constraints in the well-known game of Sudoku – a member of the group of np-complete exact set covering problems. In the context of our work, the game is used as a case study to visualize interactions between humans and computers in which the computer is an assistant to users, supporting them in understanding the problem rather than automatically providing the final solution.

## 1 Introduction

In recent years, many systems for visualizing and exploring massive amounts of data have emerged [11]. Systems such as the Attribute Explorer [13] and the EZ chooser [1] are very useful tools to help users understand the distribution of objects within a large parameter space and to select a subset of objects that satisfy user-defined parameter constraints. The systems provide users with sensitivity toward finding the most useful parameter settings by indicating objects that are barely out of bounds of selected parameter ranges, as shown in the Influence Explorer [12]. Other systems allow users to visualize and explore relationships between objects, e.g. in Cone Trees [10] and Tree Maps [5].

A topic that has not yet been investigated much concerns the visualization and exploration of constraints between objects such as relationships of joint contribution to a set, as well as relationships of mutual exclusion between objects when collecting sets. Example applications include np-complete problems in operations research, such as route planning, scheduling [15, 14], knapsack and set covering.

In this paper, we present concepts toward visualizing mutual constraints between objects in the well-known game of Sudoku – a member of the group of np-complete exact set covering problems. Sudoku puzzles impose geometric constraints on the positioning of digits on a board: every row, every column and every sub-block must contain every digit exactly once. As a consequence, digits that are already placed on the board impose constraints on positioning further digits. At the same time, every row, every column and every block requires the placement of one of each digit.

---

\* e-mail: klinker@in.tum.de

We use 9x9 Sudoku puzzles as suitable examples to investigate the underlying more general concepts of helping users visualize and explore the constraints between different settings in rows, columns and blocks of matrix-like arrangements. In principle, Sudoku puzzles can be solved automatically [7], but that is not the point. In the context of our work, the game is used as a case study to visualize interactions between humans and computers in which the computer is an assistant to users, supporting them in understanding the problem rather than automatically providing the final solution.

## 2 Physical Metaphors for the Visualization of Geometric Constraints

In this paper, we explore options (design dimensions) for visualizing constraints of mutual exclusion and set covering as physical metaphors of repulsive and attractive forces. One dimension involves the use of color, showing each digit in a different color, vs. a grayscale presentation, showing all digits in black. Another dimension explores 3D visualization schemes in comparison to 2D schemes to augment the board with visualizations of repulsive and attractive forces. This section presents the underlying concepts of the 3D visualization schemes. Corresponding 2D schemes are derived by projecting the 3D schemes onto the 2D plane that represents the game board.

Figure 4a,b<sup>1</sup> shows the arrangement for a 9x9 Sudoku board. For later reference, rows are labelled A-I, columns are labelled 1-9, and blocks are labelled I-IX. As the major means for interacting with the Sudoku game, users can place and move tiles with digits 1-9 next to the board [8].

### 2.1 Repulsive Forces (Shadow-Casting Towers)

*Towers:* Digits that are already placed on the board inhibit other cells in the same row, column and block to accept the same digit. We visualize such repulsive forces by raising respective digits on the board to high towers that cast colored shadows on the row, column and block that they are occupying (Fig. 4c).

By placing tiles to the right or below the board, users can inquire which rows, columns and blocks have already been set. The system groups such inquiries into threesomes of rows and columns in accordance with the blocking structure and it raises the respective cells on the board. The rationale for this visualization is based on one of the prime strategies for solving Sudoku puzzles: if two columns already contain a specific digit, the placement of this digit in the third column is constrained to lie in maximally three grid cells of one specific block. The same applies to threesomes of rows. In a typical interactive session, users quickly move interactive tiles up and down along the sides of the grid, thereby causing towers to rise and shrink down rapidly and providing the user with animated hints to see relationships between grid cells that have the same numbers. Judicious placement of tiles along rows and columns can exhibit unique solutions for the positioning of tiles on the board (see Fig. 4d).

---

<sup>1</sup> color figures on special color page

## 2.2 Attractive Forces (Funnels and Overview Panes)

A further important Sudoku constraint requires that each row, column and block must, in fact, contain every digit. As a consequence, rows, columns and blocks that are missing certain digits are attracting these. There are several ways of visualizing such attractive forces.

*Funnels:* Analogously to the visualization of repulsive forces, a metaphorical funnel symbolizes the fact that a cell in a respective row or column is open toward being set to the digit of the tile. Funnels are shown on all those cells of a row or column that are not yet set and that do not have a conflict with already existing digits in other rows or columns (Fig. 4e).

*Overviews of Unsatisfied Requirements:* In addition to indicating missing digits for individual cells as funnels, we also provide row-wise and column-wise overviews of all digits that have not yet been placed in a row or column (Fig. 4f). These overviews are shown on slanted two-dimensional panes hinged to the left and to the top side to the board. Each overview row/column represents a vector of all possible digits 1-9. Digits that have not yet found a place in this row/column on the board are shown, whereas those digits that have already been taken show an empty slot.

The overview panes help users quickly obtain an understanding a) of how individual digits spread across rows and columns, b) their row-wise and column-wise distributions, and c) the overall availability of individual digits. Users can see at a glance which digits have only few left-over representatives. Starting with these is a good strategy because placement of the remaining few representatives is likely to be much constrained - and thus generally easy to resolve uniquely.

## 2.3 Individual Markups

Another technique that is much used in real Sudoku puzzles concerns the systematic cell-by-cell evaluation which digits can be placed in that cell without conflict. Typically, results are penciled as tiny annotations into the grid.

Computer-based Sudoku games are very suitable for supporting such annotations. For a selected digit, the user can move the mouse across the board and explore, based on repulsive and attractive forces, whether a particular cell can take the digit or not. If yes, this can be annotated in the cell via a right mouse click. The cell then shows a miniaturized version of the tile in one of nine dedicated positions (Fig. 4g).

## 3 Implementation

As a long-term goal, the system is intended to be used for investigations whether and in which way 3D visualizations can help users explore mutual constraints between objects across a range of interactive environments [6, 9].

In this paper, we focus on the purely virtual, desktop-based setup. It operates with traditional keyboard and mouse input. All visualizations are shown on the computer monitor. A description of a Sudoku grid is read from a file, and the user solves the

puzzle in the purely virtual setting (Fig. 4). The system is designed as a model-view-controller architecture, thereby allowing for easy interchange of display and interaction schemes in the future.

There are several different groups of interactions. a) facilities for simple system control, b) schemes to manipulate the board c) methods to trigger and manipulate interactive visualizations, and d) interactions to really play the game. For each group, we consider, how control can be exerted in the current virtual setting.

For *system control*, simple toggling operations allow users to change the viewing state of the system. These operations are controlled via keyboard and mouse input.

*3D manipulation of the board*, is enabled when the the user clicks outside the interactive area of the board. 3D rotation is based on a needle map metaphor that represents vector orientation on a Gaussian or virtual sphere [4, 2].

For *interactive control of the visualization schemes*, the virtual board is extended on the right side and in the front by an interactive area. As presented in section 2, users can visualize repulsive and attractive forces by moving tiles below or on the right side of the Sudoku grid. New tiles are entered into the visualization by pressing the respective key 1-9. A virtual tile is then placed at the current mouse position. Multiple tiles of the same, as well as of other digits can be entered this way. A tile can be moved by pressing and dragging the left mouse button in the vicinity of the tile. A tile is removed from the list of tiles by double-clicking or by dragging it off the interaction area. Mockup annotations are accepted via clicks of the right mouse button. The markup status is toggled from *good* to *bad* to *undetermined* and back to *good*.

To actually *play the game*, two actions can be performed: digits can be placed on the board, and they can be removed. Users can place digits on the board by clicking with the left mouse button in an empty cell. Digits that are not declared *final* can be removed from the grid by clicking on the cell again with the left mouse button.

## 4 Results

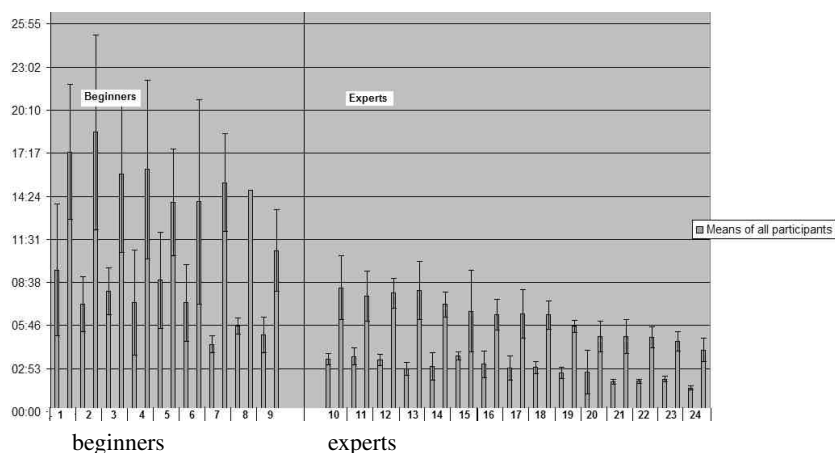
The system was implemented and demonstrated at a conference. According to feedback, we started a series of evaluations to address a number of questions that were raised and that developed further over time.

### 4.1 Colored vs. Black-and-White, 2D vs. 3D Visualizations

People questioned the value of three-dimensional visualizations for puzzles that were inherently two-dimensional. Suggestions were made to use color rather than the third dimension to emphasize attractive and repulsive forces. To address this question, we designed a test with 2 factors, each consisting of two conditions and thus resulting in  $2 \times 2 = 4$  conditions in total. We focused on the visualization of repulsive forces (i.e., shadow-casting towers), ignoring funnels and overview panes. Tests were conducted for an easy and a moderately difficult puzzle. Different variations were generated as of permutations of rows, columns and digits that did not conflict with the blocking structure of the puzzle. Figure 5 shows example pictures from all eight test cases.

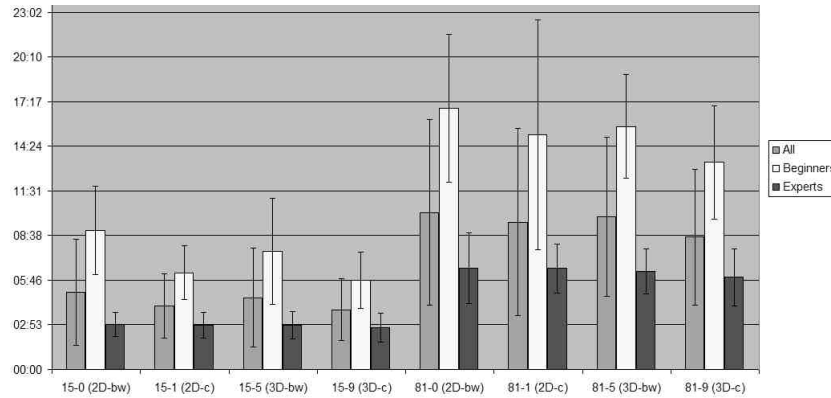
24 subjects participated in the evaluation. Age ranged from 16 to 66, with an average of 32.5 (stdev 12.9). Self-rated expertise (1=beginner, 4=fanatic) averaged at 1.9. There were 4 female and 20 male participants. Nobody was colorblind. We used a within-subject design, i.e., every subject solved all eight puzzles. The four conditions were permuted to counterbalance learning effects and fatigue. Upon arrival, subjects received an introduction to the SudokuVis program and played with it until they felt ready for the test. Test sessions lasted between 22 minutes and 2 hours.

Figure 1 presents the speed of individual participants for solving the easy and medium puzzle, averaged across all 4 conditions. It shows that there was a large variance in user expertise, ranging from a maximum average time of 18:41 minutes for a beginner's involvement in the medium puzzles to a minimum average time of 1:35 minutes for an expert (and computer game player) to solve the easy puzzles. We therefore divided the subjects into two groups, 9 beginners and 15 experts, shown in Fig. 1.



**Fig. 1.** Test results from 24 participants: average speed per participant, shown interleaved for easy and medium puzzle.

Figure 2 shows the test results with respect to the four conditions (2D-bw, 2D-color, 3D-b2, and 3D-color), both for the easy puzzle (left) and for the medium puzzle (right). The diagrams indicate a number of trends: Colored Sudokus can be solved in less time than black-and-white ones. This is particularly true for beginners (white bars) and, to a lesser degree, for all subjects combined. Similarly, 3D visualizations provide a subtle, yet consistent speed improvement over 2D visualizations. Overall, combining both factors, improvements of colored 3D visualizations over black-and-white 2D visualizations are noticeable. Yet, significant differences (using pairwise t-tests at significance level 0.05) could only be established at the beginners's level for differences between 2D black-and-white and 2D colored visualizations (2D-bw vs. 2D-c) and for 3D colored visualizations (2D-bw vs. 3D-c) for the easy puzzle.



**Fig. 2.** Test results (average times), considering all 4 conditions separately.

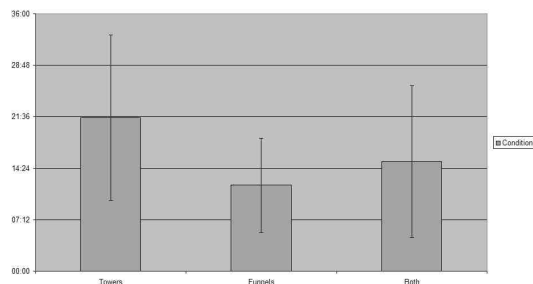
Remarkably, expert users seem to be oblivious to the choice of visualization scheme. Interviews with subjects unveiled a number of reasons. First of all, the tasks were too easy. The time to solve a puzzle was so short that it was dominated by secondary interaction tasks, such as placing and moving tiles on the board, and – more time consumingly – removing them before working on another digit. Furthermore, much time was wasted toward the end of a game just idling through all digits in linear order 1..9 to find the next digit that was not yet completed.

Subjects made another very important remark: When solving the game, they did not look directly at the towers and shadows themselves but rather at the holes on the board that they did not cover. Of course, such visual search strategy will be mostly independent of the color and three-dimensional appearance of repulsive forces.

## 4.2 Repulsive versus Attractive Forces

In a further test, we increased the level of difficulty, using a hard puzzle. Furthermore, we added overview panes to the board visualization such that users would not waste time idling through numbers to find the next promising digit. The key issue (factor) to be addressed was the finding of the previous test that users look at holes, which SudokuVis shows as funnels. We thus set up the test to investigate whether repulsive forces (towers) or attractive forces (funnels) are better suited to guide users. We investigated three conditions: exclusive visualization of towers, exclusive visualization of funnels, and joint visualization of both. Figure 6 shows example pictures from all three test cases.

Five subjects, age 20 to 50 (avg 37.2, stdev 15.7), participated in the evaluation. Two were male and three female. Self-rated Sudoku expertise averaged at 2.6. Computer expertise – especially 3D interaction – averaged at 2.4. Yet, two subjects considered themselves to be novices of highly interactive computer use, whereas two other subjects are avid computer gamers. We used a within-subject design. The three conditions were permuted to counterbalance learning effects and fatigue. Subjects were able to familiarize themselves with the SudokuVis program, using a number of training sets.



**Fig. 3.** Test results (average times) for use of towers versus funnels versus joint use of both.

The diagram in Figure 3 shows that the visualization of repulsive forces (towers) in itself is much less effective than the combined use of towers and funnels. The sole use of funnels yields the best results. Due to extraordinary differences in computer experience, test results show large variances. Thus, pairwise comparisons did not yet yield significant results.

## 5 Discussion

During debriefing sessions, several subjects emphasized that they preferred the funnel-based visualization because it did not clutter up the board as much as the shadows. Because of to this, several subjects also disliked the joint visualization of funnels and towers. Another plus of the funnel-based visualization is the fact that it focuses positive (attractive) attention on slots on the board that are under investigation. The towers, in contrast, represent negative attention. Thus, users have to mentally complement what they see with what is missing. The same is true for planning next interaction steps. For the tower-based visualization, tiles have to address rows and columns that already contain the digit in question. This does not work well with the overview visualizations – since these indicate places where digits are still missing.

Another advantage of funnels became clear when subjects started placing several tiles of different digits next to a row. This translated into a visualization request to see, for all selected digits, all places that they could currently occupy within a row or column or block. The result is equivalent to the analysis of up-to-date penciled-in markups on the board (see section 2.3), but can be called-upon much more conveniently.

On the other hand, towers were considered useful for providing a better understanding (rationale) why certain positions were not available.

In summary, towers and funnels complement one another similarly to the concepts of attribute visibility (towers) and object visibility (funnels) [11]: while towers focus on providing overviews and wide-range reasoning about the relationship between existing objects, funnels help users investigate specific slots on the board in detail (e.g., by seeing several funnels side-by-side on one slot). As a consequence, we expect funnels to be important for the forward investigation/exploration of a problem space, e.g. by helping to decide on the next step. Towers are expected to be more helpful for backward reasoning, e.g. when an error has occurred and conflicting influences exist.

We consider the SudokuVis system to be a very promising start toward visualizing and exploring complex relationships between objects.

## Acknowledgments

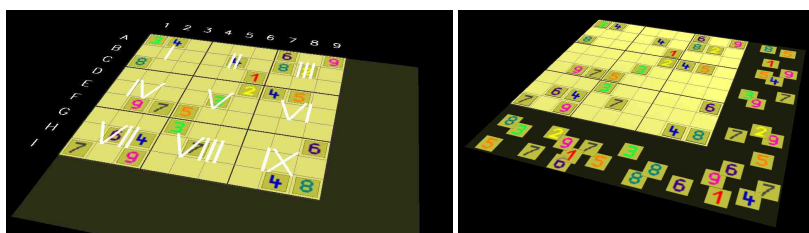
The author would like to thank Florian Echtler, Manuel Huber and Simon Nestler for their many very fruitful contributions and ideas toward playing Sudoku games on various platforms, ranging from multitouch tables to mobile phones. Further thanks go to all those anonymous test persons who spent hours playing Sudoku games in various test conditions. Finally, the author would like to thank Christoph Putz for his patience and enthusiasm in running many of the experiments.

## References

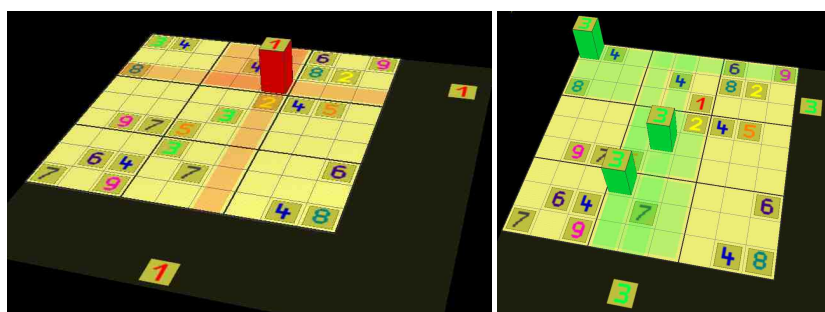
1. M. Apperley, R. Spence, and K. Wittenburg. Selecting one from many: The development of a scalable visualization tool. In *HCC '01: Proceedings of the IEEE 2001 Symposia on Human Centric Computing Languages and Environments (HCC'01)*, page 366, Washington, DC, USA, 2001. IEEE Computer Society.
2. D. A. Bowman, E. Kruijff, J. Joseph J. Laviola, and I. Poupyrev. *3D User Interfaces: Theory and Practice*. Addison-Wesley/Pearson Education, 2005.
3. G. Greenspan and R. Lee. Web sudoku, 2005. <http://www.websudoku.com>, evil puzzle 4,006,671,090.
4. K. Ikeuchi. Recognition of 3-d objects using the extended gaussian image. pages 595–608, 1981.
5. B. Johnson and B. Shneiderman. Tree-maps: A space filling approach to the visualization of hierarchical information structures. In *Proc. 2nd IEEE Conference on Visualization*, pages 284–291. IEEE, 1991.
6. G. Klinker and F. Echtler. 3d visualization and exploration of relationships and constraints at the example of sudoku games. Technical Report TUM-I0722, Technische Universität München, Department of Computer Science, Nov. 2007.
7. D. E. Knuth. Dancing links. In J. W. J. Davies, B. Roscoe, editor, *Millennial Perspectives in Computer Science*, pages 187–214. Houndmills, Basingstoke, Hampshire: Palgrave, 2000.
8. R. Lee and G. Greenspan. Jigsawdoku, 2007. <http://www.jigsawdoku.com>.
9. S. Nestler, F. Echtler, A. Dollinger, and G. Klinker. Collaborative problem solving on mobile hand-held devices and stationary multi-touch interfaces. In *PPD'08. Workshop on designing multi-touch interaction techniques for coupled public and private displays*, May 2008.
10. G. Robertson, J. Mackinlay, and S. Card. Cone trees: Animated 3d visualization of hierarchical information. In *Proc. ACM Conference on Human Factors in Computing Systems (CHI'91)*, pages 189–194. ACM Press, 1991.
11. R. Spence. *Information Visualization: Design for Interaction 2nd Ed.* Pearson Education, 2007.
12. L. Tweedie, R. Spence, H. Dawkes, and H. Su. The influence explorer. In *CHI'95: Proceedings of Sigchi*, pages 129–130, Denver, CO, USA, 1995. ACM.
13. L. Tweedie, R. Spence, D. Williams, and R. Bhogal. The attribute explorer. In *CHI'94: Proceedings of Sigchi*, pages 435–436, Boston, MA, USA, 1994. ACM.
14. P. Zhang. An image construction method for visualizing managerial data. *Decision Support Systems (DSS)*, 23:371–387, 1998.
15. P. Zhang and D. Zhu. Information visualization in project management and scheduling. In *Prod. 4th Conference of the International Society for Decision Support Systems (ISDSS'97)*, Switzerland, 1997. Ecole des HEC, University of Lausanne.



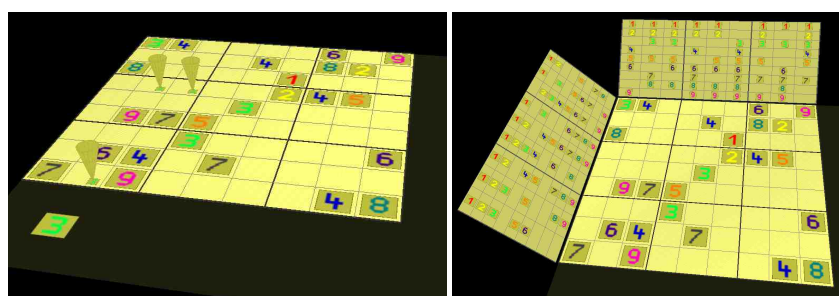
### Color figures



a) Layout of a 9x9 Sudoku board [3]. b) Arrangement of tiles next to the board.



c) Digit 1 exhibits a repulsive force on the cells in its row, column and block. d) Cell B6 must be a 3, as shown by the judicious placement of two tiles next to rows A-C and columns 4-6.

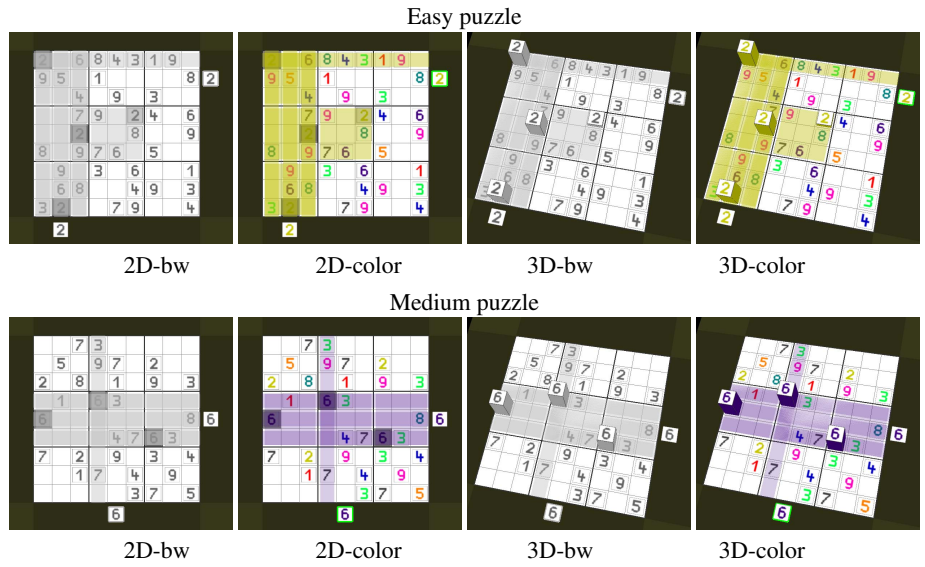


e) Cells D3 and I2 must be 3's. f) Row I still requires digits 1, 2, 3, 5 and 6.

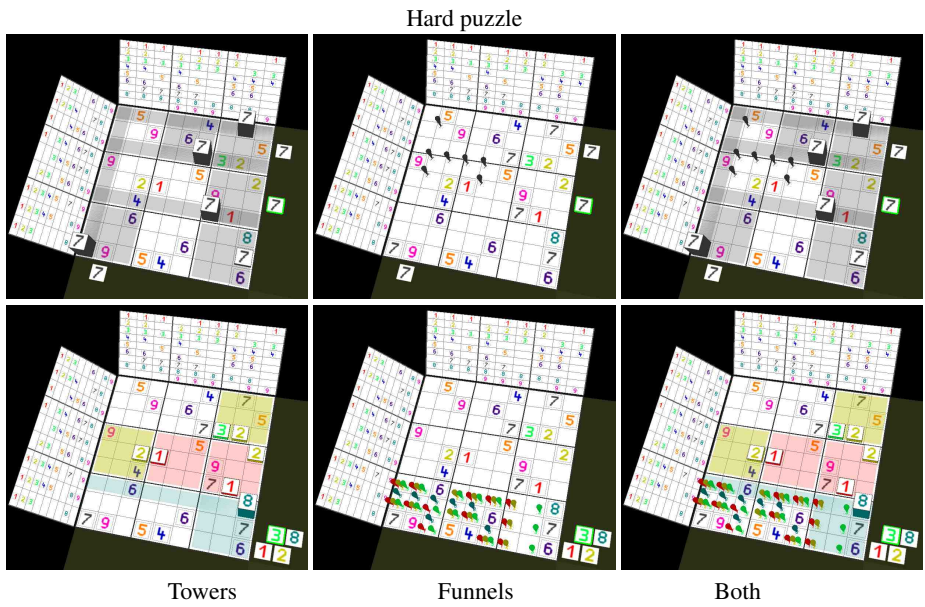


g) Visualization of a "bad" markup (darkened digit): Cell D9 cannot be a 3.

**Fig. 4.** Visualization schemes.



**Fig. 5.** Test images to evaluate the use of colors and the use of 3D.



**Fig. 6.** Test images to evaluate the use of towers and funnels.