Vismon: Facilitating Risk Assessment and Decision Making In Fisheries Management

Maryam Booshehrian, Torsten Möller, Randall M. Peterman, Tamara Munzner

1 Introduction

Fisheries management involves the regulation of fishing to balance the interests of groups of people who want to catch fish now, and the goal of sustaining viable fish populations for the future. Policy makers make these regulatory decisions in consultation with fisheries scientists, who run extensive computer simulations informed by real-world data. The sophistication of these simulations has steadily increased, and they now generate complex multi-dimensional datasets. The latest simulation models are stochastic and time-dynamic and reflect the variation in environmental influences and uncertainties of many processes [WM04]. The greatest need in this domain is to analyze the relationship between the simulation inputs and outputs. That is, there is a need to quantify tradeoffs among the simulation outputs in the form of outcome indicators, which are produced by the set of management options that are part of the simulation inputs. A second goal is to support sensitivity and uncertainty analysis; that is, to understand when small changes in inputs lead to relatively large changes in outputs.

However, the sophistication of visualization support for the analysis of, and communication about, these model outputs lags far behind, leaving an unmet need for more effective interactive visualization tools. Most analysis is done with simple individual plots: viewing time series plots with multiple curves, individual scatterplots to see correlations, and contour plots for overview summary information. The need to link between these plots has long been recognized; decades ago one of us (Peterman) proposed doing so manually with carefully aligned paper printouts of multiple plots and physical transparency printouts with multiple crosshairs [Pet75]. Obviously, these manual methods are inadequate for the complex datasets of the present.

The contribution of this paper is two-fold. We present a data and task analysis of the fisheries domain, developed in close collaboration with fisheries scientists. We also present the design and implementation of the Vismon system, an interactive visualization tool that supports sensitivity and uncertainty analysis through multiple linked views.

This design study paper continues with background information about fisheries policies and science in Section 2, followed by the fisheries task and data analysis in Section 3. We continue with a description of the Vismon design and interface in Section 4, and a walkthrough of Vismon with a case study in Section 5. We conclude with a discussion of future work in Section 6. Vismon is a work in progress; this interim technical report does not include a discussion of the related work in visualization or a full evaluation of whether its design goals are met.

2 Fisheries Background

We begin with background information about the policy making context of decision-making support for fisheries, and continue with a summary of the scientific problems they face.

2.1 Fisheries Policy Making

Fisheries management is in the midst of major changes, both in terms of the policy making situation and in terms of physical and biological conditions. For setting policy, there is greater stakeholder involvement and a stronger expectation of accountable decision making than in the past. Institutional roles are also shifting. Fisheries science was once the uncontested domain of the government and academia, but is now being undertaken by other groups including non-governmental conservation groups and the fishing industry itself. Physically, the impact of climate change means that the processes being studied have moving baselines. Efforts to interpret cause and effect are often confounded by the reality of multiple simultaneous changes in both natural and human processes.

The goals of policy makers in fisheries management have evolved over time, in three major waves. The first was a *command and control* model: "we have the data, and we decide". The second model, *decide, announce, defend*, grew out of a push for public consultation, but that did not go far enough for meaningful input from the public. The current model is *multi-stakeholder consultation* with the new idea of including many stakeholders in the analysis process itself. Thus, the current goal of policy makers in fisheries management is to make well-informed decisions based on large amounts of quantitative information. The new need is to support not only communication to, but also analysis with, multiple stakeholders.

The three roles of interest in this process are fisheries managers, stakeholders, and fisheries scientists. The fisheries scientists need to provide fisheries managers with extensive quantitative information to support decision making, via simulation informed by monitoring efforts. The managers choose actions to best meet management objectives, such as maintaining sufficient spawning fish and allocating the allowed catches among the competing interest groups. The stakeholders include environmentalists and three different fishing interest groups: the commercial fishing industry, subsistence fishing communities (both First Nations and other groups), and recreational anglers.

2.2 Fisheries Science

The process of fish stock assessment has the goal of quantitatively estimating potential outcomes of contemplated management options [Pet09]. Difficulties arise from both our limited understanding of the structure and function of aquatic systems, and the uncertainties inherent in the data given current monitoring capabilities. The specific challenges of assessment include at least four factors [Pet04]. First, there are pervasive uncertainties and risks. Second, there is the challenge of estimating probabilities for uncertain quantities. Third is the problem of evaluating the performance of management options. The final challenge is communicating complex technical information to decision makers and the public; that is, conveying assumptions, results, and implications to people not actively involved in the analyses.

There are several different sources of uncertainty and risk. Aquatic ecosystems are variable because they are natural environments. A stochastic dynamic model of a natural population accounts for this variability by implicitly encoding a range of alternative hypothesis about possibilities. Moreover, only imperfect information is available to scientists. Real-world data collection inevitably involves observation and sampling error, which must also be incorporated into the stochastic model. Governmental control of human behavior is also imperfect. There may be imperfect compliance with regulation by harvesters, and there is natural variation in catchability. The term *outcome* or *implementation uncertainty* is used for these factors. Another source of uncertainty is a lack of clarity about management objectives; there may be no consensus on the desired outcomes.

Simulation in service of stock assessment has nevertheless been heavily used for decades, with increasing complexity and sophistication [Pet75, Pet04, Pet09]. Peterman describes a high-level risk assessment framework to use when addressing these complex problems through simulation [Pet04]. The first step is to define management objectives along with indicators that can measure how well they are met. Examples of indicators are the expected catch, the catch variation over time, or the spawning biomass. Indicators might have an associated undesirable probability of falling under some critical limit. The second step is to consider several management options explicitly by stating them as inputs to the simulation model. The third step is to run the stochastic model with a wide range of hypotheses expressed as parameter values and alternative submodels used inside the simulation. Finally, a sensitivity analysis can be done to see the effects of the different assumptions on the outcomes.

3 Data and Task Analysis

We now discuss the data, workflow, and tasks for this design study, moving from the domain-specific details to the abstractions that we have chosen.

3.1 Data Abstraction

The data abstraction that we use is that there are two independent input dimensions to the simulation. These are known as the **management options**, or **options** for short. Our focus is on these user inputs. There are many other inputs to the simulation model that are not exposed to the user of Vismon. They are used to estimate the parameters for functional relationships assumed to link the processes within the model; for example, the relationship between sea surface temperature and mortality rate. There are many dependent output dimensions, known as **indicators**. The simulation generates the outputs given the inputs by running hundreds of Monte Carlo simulation trials. Each indicator can be summarized by a few statistical measures, but the full underlying dataset is also available.

This abstraction covers a significant and interesting part of the possible design space, but not all of it. In current fisheries practice, most simulations do not exceed two independent input dimensions, as was the case for our collaborators as well. Many simulations use only a single one, a case that Vismon handles easily. While the future will clearly bring simulations with three or more independent dimensions into the mainstream, we chose two as a good place to start because it is richer than the very simple case of one but steers clear of the complexities of many. The design target for number of output dimensions was 10 to 20, again motivated by the current needs of our collaborators, which reflect the multiple indicators of several groups of stakeholders.

In this paper we will use a specific model as a concrete driving example: the chum salmon populations in the Arctic-Yukon-Kuskokwim (AYK) region of Alaska, USA [CPZ09]. The motivation for this particular model was concern with the large and rapid decreases in abundance in the late 1990s. The three major stakeholder interests at play in this region are sustainability, commercial fishing revenue, and subsistence fishing. The AYK chum salmon simulation has datasets for five different rivers.

One input option reflecting managers' objectives is the **escapement target**, which is the desired number of spawning salmon; that is, the number that "escape" being fished. The other is the **harvest rate**, the number of fish that the combination of the commercial and subsistence harvesters should catch *after* the escapement target is met. Each option is set to one of 11 levels, so the simulation covers a total of $11 * 11 = 121$ combinations of these input parameters. Each of these combinations is called a *scenario*.

The simulation output is 12 indicators for each scenario, grouped into 3 categories: escapement, subsistence catch, and commercial catch. Each run of the simulation covers a 100-year time period, and the indicators are statistical measures to characterize the results in each category with four output numbers: the average, median, temporal coefficient of variation (standard deviation divided by the average), and a risk measure expressed as the percentage of years that something undesirable happened during the hundred simulated years. The risks for each category in turn are when the run size is below the escapement target, when the subsistence fishery is below the lower quartile of historical catches, or when no commercial fishing is allowed.

In fisheries science, indicators of outcome need to be either maximized or minimized, depending on their role in the analysis. As an example, the commercial catch should be increased, but the probability of over-harvesting needs to be reduced. There are no cases in which an indicator should have both minimum and maximum limitations. Each of the twelve indicators thus has a direction of desired change as associated metadata, in addition to its set of quantitative values. The simulation output includes both average and median so that the analysts can easily check whether these quantities are similar, indicating normal distributions. When they are different, their relative values give a quick sense of the direction of the distribution's asymmetry.

The stochastic simulation carries out 500 Monte Carlo trial runs for each scenario. The full dataset has 726,000 data elements in total, from the product of 121 scenarios, 12 indicators, and 500 runs. This dataset is unwieldy enough that a simplified high-level dataset is also computed by aggregating the values over the 500 runs into a single number, for a dataset of only $121 * 12 = 1452$ elements. (This number can be either the average or the mean.) We use the term **underlying uncertainty** to mean the information contained in the full Monte Carlo output dataset that is available only in aggregate form in the high-level dataset; that is, the 726,000 elements as opposed to the 1452 elements.

3.2 Task Abstraction

As we discussed in Section 1, Vismon was designed to help quantify tradeoffs that result from choosing different management options, and to support sensitivity and uncertainty analysis, namely to understand when small changes of input lead to large changes of output rather than small ones.

We now break this problem down into a more detailed list of subtasks:

1. summarize a large number of simulations,
2. add constraints on the ranges of values for the simulation input (management options) and output (indicators) based on stakeholder interests,
3. select a few candidate combinations of options,
4. quantify tradeoffs between selected options,
5. avoid sensitive regions of the parameter space where indicator values change rapidly per unit change in a management option,
6. avoid options with high underlying uncertainty across the 500 Monte Carlo trials,
7. enable communication among scientists, policy makers, and stakeholders.

We abstracted this set of domain-specific subproblems into a smaller set of generic tasks:

- narrow down from a large set of possible scenarios to a small set of candidates,
- compare a small set of scenarios,
- avoid uncertain scenarios,

and a fourth task that crosscuts these first three:

- facilitate communication between technical and non-technical people.

3.3 Previous Workflow

At the start of our collaboration, Peterman's group was actively engaged in the analysis of simulation results, as they had been for years. Their previous analysis procedure was to use a wide range of individual plots generated with scripts for R and other similar packages. Although scripts for general-purpose frameworks are a powerful and flexible way to create nearly any individual view showing details at a low level, they require the user to know exactly what to specify in advance.

The major problem was that only a tiny fraction of the information theoretically available in the dataset was actively considered in the analysis process. The scientists and managers were buried by the quantity of information put out by the models. Essentially, they picked a few points in the parameter space through trial and error and ignored the rest, because they did not have a systematic way to explore the information. Their view of the dataset was narrowly focused; they lacked high-level overviews and other ways to easily synthesize information across a combination of low-level detail views. Exploring the dataset at the level of the aggregate statistical measures was very difficult, and understanding the underlying uncertainty expressed in the full details of the Monte Carlo runs was even more so.

Their analysis procedure was most successful in supporting the first two subtasks at a basic level: summarization and adding constraints. A very small set of candidate management scenarios was picked (Subtask 3) based on past experience, rather than through a data-driven exploration of the simulation output. While the high-level tradeoffs were well known to the scientists and managers, quantifying them for any specific combination of choices was difficult because the relationships are nonlinear (Subtask 4). Quantifying tradeoffs involved a great deal of cognition and memory, with only minimal help from their perceptual system, in order to synthesize information across multiple individual views. Avoiding sensitive regions (Subtask 5) required a great deal of trial and error. Inspecting an individual contour plot showing the values for one indicator could show them regions of rapid change for that indicator where the contour lines were closely spaced, but synthesizing a mental model across all of the indicators was not well supported by the available methods of analysis. Understanding the complexity of the Monte Carlo trials (Subtask 6) was not easily addressed; the scientists typically just worked with the averages because the full dataset was too overwhelming. Communicating results (Subtask 7) was only partially addressed. Although their process did support some level of communication between scientists, simulation results were very difficult for policy makers to understand and extremely challenging for stakeholders and the public to grasp.

Considering their tasks at a generic level, we conjectured that many useful scenarios might not even be considered in the candidate set, and conversely that too much analysis time was being spent exploring candidates later found to be unsuitable. Our goal was to allow the scientists to greatly increase the breadth and scope of their analysis, even while reducing the total time required.

3.4 Design Requirements

We identified three major design requirements.

1. Speed up and extend their previous analysis workflow:
   - Provide interactive linking and brushing across multiple views.
2. Add new capabilities for risk assessment analysis:
   - better support for detailed tradeoff analysis,
   - make the underlying uncertainty in the data visible but do not force uncertainty analysis on users who want to start simply,
   - allow the three generic tasks to be done either in order or interleaved.
3. Support easier interpretation for policy makers and the public in addition to supporting scientists in the analysis process.

A clear starting point based on Requirement 1 was to build an interactive visualization system with multiple linked views using their familiar and effective visual encoding technique of contour plots. It was obvious to us from known visualization design principles that this baseline capability would speed up the previous analysis process immensely, since they were essentially doing linking and brushing by hand.

Requirement 2 encapsulates many of the interesting visualization research questions beyond the obvious baseline, and the design decisions arising from them are discussed in the next section.

Although there is a certain level of dependency between the three generic tasks, one design goal was to allow these tasks to be fully interleaved, so that uncertainty information could be incorporated into the selection of the candidate actions rather than being used only after narrowing down to a handful of options, as was the case with their previous workflow. On the other hand, the requirement of encouraging but not forcing uncertainty analysis led us to the design goal of creating views that could all be used in a straightforward way with only the information from the high-level simplified dataset. The combination of these two goals led us to a strategy where views could be augmented with information from the full underlying uncertainty dataset at different levels of complexity. For example, a user should be able to start exploring scenarios using only the high-level average values, and then later incorporate uncertainty information to see which of them are uncertain or risky. The user should also be able to include that information in the initial exploration, so that when faced with scenarios that have the same average values, they can prefer the more certain ones.

Requirement 3 led us to a multi-stage iterative refinement and evaluation strategy. The first stage is to work closely with the scientists to ensure that the tool does support and extend their analysis process. We have done so; this technical report documents that first stage of the work. The second stage is to deploy the tool to policy makers to see that they can also effectively use the tool for analysis, and refine it as needed. We are currently in the midst of this stage. The final stage will be to test that policy makers can indeed use the tool to effectively communicate with the general public, which we plan to do as future work.

4 Vismon Interface

Vismon is built using multiple linked views, as shown in Figure 1. The three main data abstractions used in Vismon are options, indicators, and scenarios. Each of the three main views has a different visual encoding to emphasize different aspects of these elements and the relationship between them, but the color coding for scenarios is the same across all of them. Each main view is itself composed of small-multiple charts, with linked highlighting between analogous items on mouseover.

The Overview pane on the left has sliders that show the

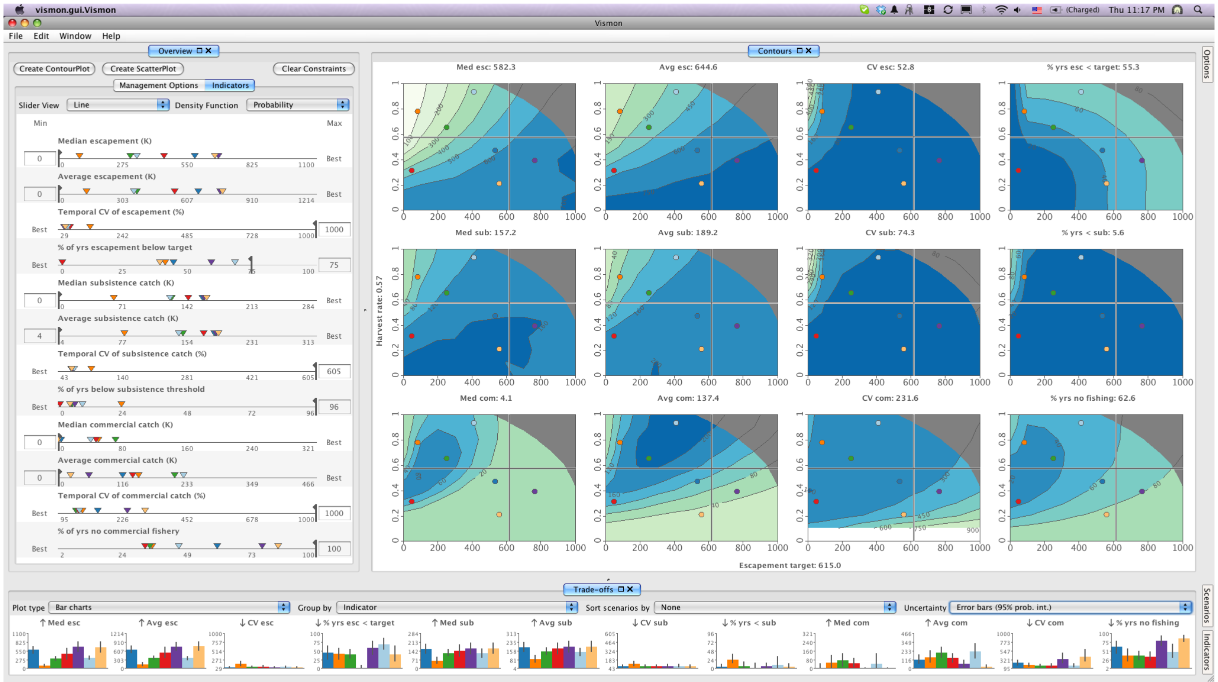

Figure 1: Vismon interface: Overview pane, top-left, shows the list of management options and indicators in separate tabs; Contour Plot Matrix pane, top-right, shows the contour plots of indicators over the two management options and supports scenario selection; Trade-offs pane, bottom, shows detail with the indicators for the selected scenarios.

range for, and allow constraints to be imposed on, the individual options and indicators, with optional histograms showing the underlying data distributions for a richer view. The location of each scenario with respect to the value ranges is shown with a colored triangle. The Contours pane on the right has a contour plot matrix with one plot for each active indicator showing the values along the options axes, with scenarios shown as colored dots. The Trade-offs pane on the bottom shows details about the indicator values for the active scenarios through bar charts, star glyphs, or multipodes plots. The view can either show a chart for each scenario with marks for the indicators, or vice versa.

Constraints set in the Overview pane immediately change the greyed-out regions in the Contours pane that indicate unacceptable management options. Scenarios chosen by clicking on one of the 121 grid points in the Contours window appear in both other views.

4.1 Overview Pane

The Overview pane shows the individual options and indicators as one-dimensional ranges, with one tab for each type. In both cases, the base small-multiple view shows a slider, with both a moveable handle for quick interactive positioning and a text box for precise numerical entry when the user knows a value of interest in advance. The sliders allow the user to restrict the active range of any input option or output indicator, which changes the shape of the permissible scenario region in the Contours pane plots.

The results of moving the input option sliders are not surprising; a straight line sweeps out horizontally or vertically to change the rectangular size of the active region, because these values correspond with the underlying grid used for both simulation computation and the contour plot display axes. However, changing the undesirable range of the output indicators leads to complex and non-obvious shapes for the active region. With just a few minutes of exploring with these sliders, the analyst can get the gist of how constraining the different input and output dimensions affects the set of possible scenarios.

Options have bidirectional sliders with both a minimum and maximum handle, and two text boxes. Indicator sliders have only a single handle, since their directionality is known from the metadata. The label `Best` appears instead of a text box on the side that is the most desirable direction, and the handle also has a small flag pointing in that direction as a subtle visual cue.

The plain sliders for the indicators convert to scented widgets [WHA07] on demand from the user, showing histograms of distributions in the underlying dataset in order to provide more guidance on what choices to make when setting the ranges. There are two choices, either or both of which can be shown.

The simpler choice, `MC Trials`, shows a histogram with the distribution of all values for this indicator across all the Monte Carlo trials. Figure 2a shows an example for the `Median escapement` indicator, where the slider bar has been moved from the default position of 0 to the value of 319. We can see this is an indicator where the maximum value is the most desireable because the `Best` label is on the right side of the slider. The geometric intuition is straightforward: the user can see in advance whether a small or a large part of the distribution will be filtered out when the slider bar is moved to a particular position, rather than using a trial and error process where the slider is moved and then the results are scrutinized.

Figure 2b shows the result of drilling down even further by clicking on the red triangle representing a scenario. The histogram has a colored overlay allowing the user to compare the distribution of the trials just for the chosen scenario with that of the full dataset of all 121 scenarios. The `Log` label appears on the left to show that in this mode the vertical axis is now log-scale rather than linear, to ensure that the overlay details are fully visible.

The more complex choice, `Probabilistic Objectives`, allows a sophisticated user to reason about all the Monte Carlo trials, not just their average. It uses a two-part filter with a second slider and histogram. The base slider still sets a limit on the value of the target indicator. The second slider allows the user to set a probabilistic limit corresponding to the percentage of Monte Carlo simulation trials that are above that indicator limit for indicators that need to be maximized, or below for those that need to be minimized. The second slider allows the user to change this probability value interactively from the default of 0%, meaning that no possibilities have been ruled out, up to a higher number. In Figure 2c, the user has set the probability that the `Median escapement` indicator is greater than 319 to 54%. The plots in the Contours pane will show which of the 121 indicators have been ruled out by this limit by crossing them out with X's, as illustrated in the Figure 16 example.

By default all histograms show probability directly, but the user can switch them to show cumulative density functions, as in Figure 2d.

We considered a design change to compress the histograms in the Overview pane to use less vertical space, so that users could see more augmented sliders at once without the need to scroll. However, our collaborators had strong opinions that current information density is at a good set point, and that making the histograms any smaller would impede their utility.

The sliders are not only controls but also displays, even when not augmented by the histograms. They act as legends that document the range of each option or indicator; the sliders have the same visual range on the screen but cover very different regions of data space. They also show the full name for options and indicators, rather than the short names used in the other panes to save space. Most importantly, the scenario triangles show the distribution of the scenarios with respect to these ranges in a high-precision way using spatial position. That distribution would require more mental effort to glean from the Contours plots, where it is encoded more indirectly and with lower precision as the color of the contour band in which the scenario dot is embedded.

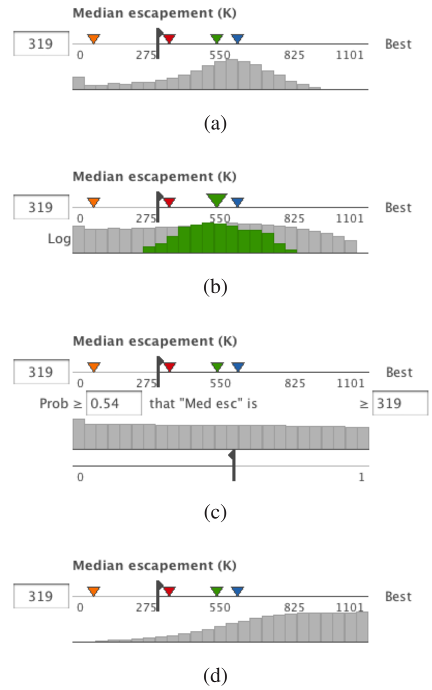

Figure 2: The Overview pane sliders become scented widgets [WHA07] on demand. (a) `MC Trials` shows the distribution of all values across all Monte Carlo trials for this indicator. (b) Selecting a scenario by clicking on its triangle shows its distribution compared to the full one across all 121 scenarios. (c) `Probabilistic Objectives` allows a second, probabilistic aspect of the indicator to be set as part of a management objective. (d) The histogram can show the cumulative density function rather than the direct probability density function.

4.2 Contours Pane

The Contours pane contains a contour plot matrix that has one two-dimensional plot for each active output indicator. Each (x,y) location in the contour plot represents a scenario. Again, a scenario is characterized by two independent variables, the parameter settings used for the input management option choices, and has many dependent variables, the output indicators. The small-multiple views are linked with a crosshair that appears at the same (x,y) location in each of them when the cursor moves across any of them, and the exact numeric value for the indicator at that point is shown in each title bar.

The plots are all linked to the Overview pane sliders that provide data-driven constraints on the active region within each of them. All plots have the same two axes of the options, and the demarcation between the colored active region and the greyed-out restricted region is the same in all. The plots show the high-level dataset: either the average or the median of the underlying 500 simulation runs for their indicator. The plots resize dynamically to fit within the pane as it resizes or the number of plots to show changes as indicators are de- or re-activated, so that they are always visible side by side without the need to scroll. By default, all indicator plots are shown; Figure 1 shows the full set of 12 in the example dataset. They can be can be turned on and off with a right-mouse popup menu when the cursor is over a plot, or through the control panel for the pane, which is accessible through the Options button on the right of the pane.

The contour plots are colored by default with a sequential blue-white colormap that incorporates hue and saturation in addition to luminance in order to make the contour patterns highly visually salient. Several other colormaps are built in, all generated by ColorBrewer [HB03]. The direction of each indicator is taken into account, so that the dark and highly saturated end is the most preferred value for each.

The static array of contour patterns provides an overview of the high-level dataset that is focused on the individual indicators. Moving the cursor across the plot allows fast comparison between the indicator values for a single scenario because of the dynamically linked crosshairs. We chose to keep a contour plot matrix at the heart of the system because they were both familiar and effective.

One of the main uses of this view is to guide the user in selecting a small set of candidate scenarios, which can be compared in detail in the Trade-offs pane. Clicking within a contour plot selects the scenario at that point. Its location is marked with a colored dot in all plots in this pane and a colored triangle along each indicator range on the Overview pane sliders. The marks representing selected scenarios are small and show the categorical data type of an identifier that is unique for each scenario, so they are coded with high-saturation colors in different hues. This shared color coding acts as a link across the different views. The design target is that users are unlikely to select more than 10 scenarios to inspect in detail at once. A palette of pre-selected highly distinguishable colors is used for the first 11 scenarios, with random colors used for any additional ones. The user can use the color picker in the Trade-off pane control panel to override these color defaults.

The user can also explore some of the underlying uncertainty data in the Contour pane, as shown in Figure 3. The 11×11 grid through which the contours are interpolated is the set of 121 pre-computed scenarios, which can be shown on demand as small points. These points can be size coded with two additional numbers that summarize the underlying 500 Monte Carlo trials in terms of the same 95% confidence interval information that is used for the error bars described in the next section. Uncertain regions are clearly indicated by large dots, and strongly asymmetric intervals can be seen where the dots have visibly different aspect ratios. The user can also turn on a histogram showing the full distribution over the 500 trials at the point under the crosshair. The histogram updates as the cursor moves, and can be displayed either in the plot containing the cursor or in all of the linked plots.

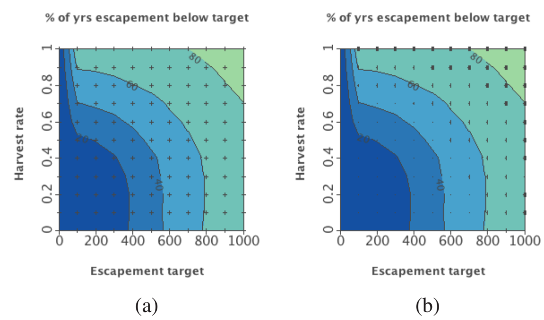

Figure 3: Contour plots can reveal the underlying 11×11 grid. (a) Small points show where the grids are; (b) The points are size coded to show the underlying uncertainty in data.

4.3 Trade-offs Pane

The Trade-offs pane, as the name suggests, supports a detailed assessment of the trade-offs between a small set of scenarios. Again, it does so with a set of small-multiple plots. The default plot type is standard bar charts.

These plots support two kinds of analysis. The default mode is to group outputs by indicator, showing one plot for each indicator with a different colored bar for each scenario. Figure 9 demonstrates how this mode allows easy comparison of how indicators change across scenarios. The opposite mode is to group by scenario, where each plot shows a single scenario with the bar heights showing all of its indicators. Conversely, this mode allows easy comparison of indicator values within a particular scenario, and the profiles of entire scenarios with each other, as in Figure 12.

The plots support four different levels of showing the underlying uncertainty information, as shown in Figure 4a-d. The simplest possibility is *none*, to show plain bars with no uncertainty information at all. This option fulfills the requirement that analysts who want to do only high-level analysis should not be forced to deal with uncertainty, and only the high-level dataset information is encoded with bar

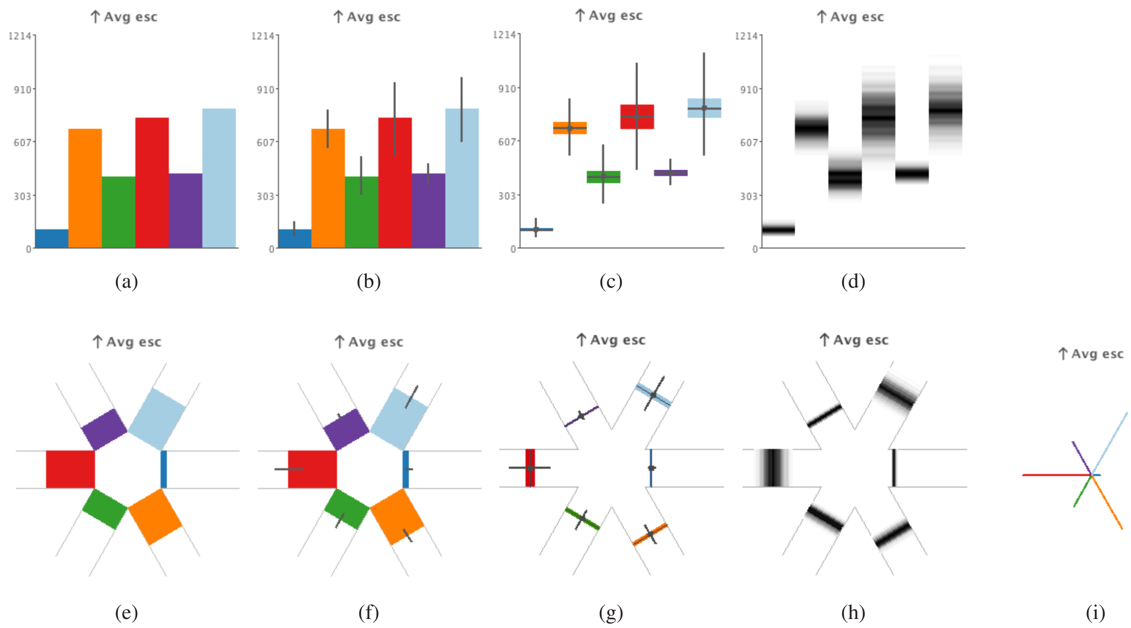

Figure 4: The Trade-offs pane can show uncertainty information on bars in four ways. **a)**: None. **b)** Error bars. **c)** Box plots. **d)** Shaded distributions. **e)** The multipodes display shows analagous information for a second plot type that uses radial bars: **e)** Radial none. **f)** Radial error bars. **g)** Radial box plots. **h)** Radial shaded distributions. **i)** The third plot type is star glyphs, with thin radial lines rather than thick bars, so uncertainty information is not shown.

heights. The default *error bar* mode superimposes a simple error bar showing the 95% confidence interval on top of the mark, summarizing the uncertainty with two additional values. We use this as a default since the idea of error bars is very familiar to most scientists, and the high-level information remains very salient. In *box plot* mode the uncertainty is shown using a stylized box plot that shows the five statistical values of minimum, lower quartile, median, upper quartile, and maximum. For visual consistency, the box between the upper and lower quartiles is filled with the same color as the bar marks, and the box whiskers and median mark are drawn in the same style as the error bars with thin grey lines. The box plot is used in science as much as error bars, and it is a part of the conventional statistical graphics toolbox. The high-level aggregate number is still shown explicitly, but with less salience. The *shaded distribution* mode shows the uncertainty information in full detail by using a greyscale map that encodes the full distribution as normalized density. The high-level aggregate number is not necessarily the most visually salient aspect of the display. This visual encoding conveys the most information, but is quite different than the other more familiar displays; the box plot acts as a bridge between the familiar and the powerful.

The Trade-offs pane provides two other possible plot types. The default choice shown in Figure 4a-d is the familiar bar chart, where one-dimensional marks aligned on a horizontal axis encode values with spatial position along a vertical axis. The **multipodes** plot in Figure 4e-h is a radial alternative where the bars are laid out along a circle. The flat layout allows more accurate comparison of bar lengths, whereas the radial layout allows easier comparison between the first and last bars. Both the standard bar chart and the radial bar chart can show the underlying uncertainty in analagous ways. The third plot type is the star glyph, a simpler version of the multipodes plot that uses a one-pixel width mark instead of a wider bar, shown in Figure 4i. Star glyphs do not support showing uncertainty information as above, but they are a visual encoding that is already familiar to many fisheries scientists. We thus include them as a bridge to the less familiar but more powerful multipodes plots.

When the plots are grouped by indicator, the bars in all of them can be sorted by the value of any indicator as shown in Figure 14, rather than the default based on the order in which the scenarios were created.

4.4 General Functionality

There are several other general features. The user can request a separate window showing a single large contour plot for any indicator. These high-resolution plots are fully linked in all of the ways that the small multiples in the Contours pane are. The user can also select any two indicators and create a separate window with a scatterplot comparing the 121 scenario values. This plot is linked to the main windows only by the color coding the dots representing selected scenarios. Although scatterplots were not heavily used in the previous workflow, they were sometimes useful for analysis of the Pareto frontier [EMKH10]. Users can add persistent reference lines that appear in all contour plots at specific values of the options. Finally, the selected set of scenarios can be exported as a comma separated value (CSV) file, so that results from a Vismon analysis session can be used elsewhere. Export of any window as a PNG image file is also supported, so that Vismon results can be easily shared with others.

Vismon is implemented in Java 1.6, with diagrams drawn using custom Java2D graphics code. The tick marks on plot axes dynamically adapt to use the space available using the algorithm of Talbot, Lin and Hanrahan [TLH10].

5 Vismon: A Walkthrough

We now present a case study of how Vismon can be used in the form of an illustrated walkthrough to show how the tool supports analysis and decision making. The target user is a fisheries manager in Alaska who hopes to make an informed policy decision by finding a scenario with low uncertainty that best suits her objectives. The dataset is the driving example described in Section 3, a simulation model of the chum salmon population in the AYK region.

Figure 5 shows the Vismon display on startup. No constraints have yet been set, so the entire rectangular region of each plot in the Contours pane on the right is fully colored to show that it is active. The active region is all of the (x,y) combinations of management options that are acceptable given the constraint limits.

The manager now decides to add constraints using the sliders in the Overview panel on the left, to reduce the size of the active region by eliminating scenarios that produce unacceptable values of particular indicators. She first chooses 50% as the maximum allowable percentage of years in which escapement is below the target, and a curvilinear region in the upper right becomes greyed out, as shown in Figure 6. She then chooses 30% as the maximum acceptable percentage of time in which subsistence catch is in the lower quartile historically, and Figure 7 shows the continuing decrease in the size of the active range. Finally, she sets a minimum of 100,000 fish for the average annual commercial catch, as shown in Figure 8. The resulting active region in the contour plots showing the feasible scenarios is much smaller than the full original set, thereby simplifying the complexity faced by the manager. Some indicators within the active region are dark blue, showing that they are in the highly preferred range, while others are the lighter green color indicating unfavorable values.

The manager then explores a few management options by clicking the mouse in a few locations within the active region, and the Vismon window updates to include the Trade-offs pane showing detailed information about those scenarios, as shown in Figure 9. The manager decides that she no longer needs to consider any of the indicators that pertain to median values. After she deactivates those with the right-mouse popup menu, contour plots use the newly available room.

Figure 10 shows the display after she switches to showing more detailed uncertainty information as box plots rather than error bars. She then digs even deeper by looking at the shaded distributions, as shown in Figure 11. She then switches to grouping by scenario rather than indicator and returns to bar charts so that she can compare the profiles easily between her four scenarios, shown in Figure 12. Figure 13 shows the display after she switches back to grouping the charts by indicator. She scrolls the Overview down to look at the commercial catch indicators, and then sorts the charts in the Trade-offs pane by the `Average commercial catch` indicator. She also changes the Overview pane settings to show the distributions for each indicator. Figure 14 shows her view after selecting the red scenario to compare its distribution to the full dataset; the histograms in the Overview pane are now log-scaled on the vertical axis so that the details are visible.

Figure 15 shows the result of turning on the histograms in the Contours pane to see details about the full Monte Carlo trials for the scenario under the crosshairs. This additional uncertainty information leads the manager to realize that the current constraint settings might need to be reconsidered. Figure 16 shows the display after she turns on the second set of histograms underneath the sliders in the Overview, and sets the probabilistic limit that the `Average commercial catch` must be more than 100,000 fish in 75% of the Monte Carlo trials. The Contours plots now have many crossed out locations, indicating the option combinations that have been ruled out by this setting for the probabilistic acceptable values.

6 Future Work

Vismon is a work in progress. It was developed through a close collaboration between visualization designers and one fisheries scientist (Peterman, a coauthor of this paper). A fully interactive first prototype was developed as a testbed, allowing us to get periodic feedback via demonstrations to Peterman and several other fisheries scientists. We also received useful responses from a group of forty scientists and policy makers from the Alaska Department of Fish and Game. This paper describes the second generation prototype, which was been fully redesigned for usability based on the successful elements of the first one. This version of Vismon is available for free download at `http://www.vismon.org`.

The next step will be to deploy this prototype to a larger audience to use as part of their daily work routine. This field trial will help us determine whether the tool meets its design goals. We will include both fisheries scientists and policy makers at the management level in this assessment.

We have thus far focused most of our prototyping efforts on capability and usability. The system achieves interactive response on fast hardware with the datasets in use by our collaborators, but can lag on older machines. We predict that significant performance improvements could be gained with a minor amount of engineering effort. We plan to do so as part of a robustness pass before the second-stage deployment of the tool.

This technical report is an interim document. We have not included a full discussion of the related work in the visualization domain to frame our contributions, nor have we fully evaluated whether our design successfully meets the needs of its target audience.

References

[CPZ09] COLLIE J. S., PETERMAN R. M., ZUEHLKE B.: *A risk assessment framework for AYK salmon*. Tech. rep., Project completion report for the Alaska Sustainable Salmon Fund, Nov. 2009. 2

[EMKH10] ESKELINEN P., MIETTINEN K., KLAMROTH K., HAKANEN J.: Pareto navigator for interactive nonlinear multiobjective optimization. *OR Spectrum 32*, 1 (2010), 211–227. 9

[HB03] HARROWER M., BREWER C. A.: Colorbrewer.org: An online tool for selecting color schemes for maps. *The Cartographic Journal 40*, 1 (2003), 27–37. 7

[Pet75] PETERMAN R. M.: New techniques for policy evaluation in ecological systems: Methodology for a case study of pacific salmon fisheries. *Journal of the Fisheries Research Board of Canada 32*, 11 (Nov. 1975), 2179–2188. 1, 2

[Pet04] PETERMAN R. M.: Possible solutions to some challenges facing fisheries scientists and managers. *ICES Journal of Marine Science: Journal du Conseil 61*, 8 (Jan. 2004), 1331 –1343. 2

[Pet09] PETERMAN R. M.: Fisheries science in the future. In *The Future of Fisheries Science in North America*, Beamish R. J., Rothschild B. J., (Eds.), vol. 31. Springer Netherlands, 2009, pp. 167–184. 2

[TLH10] TALBOT J., LIN S., HANRAHAN P.: An extension of Wilkinson's algorithm for positioning tick labels on axes. *Visualization and Computer Graphics, IEEE Transactions on 16*, 6 (2010), 1036–1043. 9

[WHA07] WILLETT W., HEER J., AGRAWALA M.: Scented widgets: Improving navigation cues with embedded visualizations. *IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis 07) 13* (2007), 1129–1136. 5, 6

[WM04] WALTERS C. J., MARTELL S. J. D.: *Fisheries ecology and management*. Princeton University Press, 2004. 1

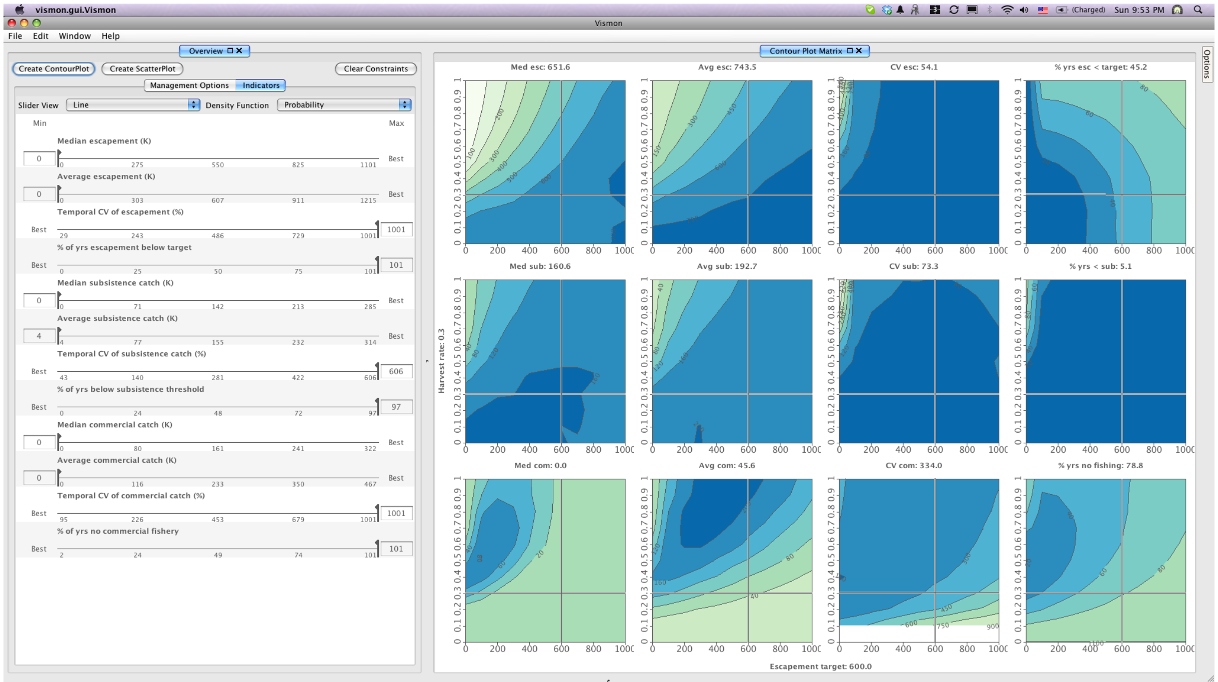

Figure 5: On startup, the active region of the contour plots is the full rectangle.

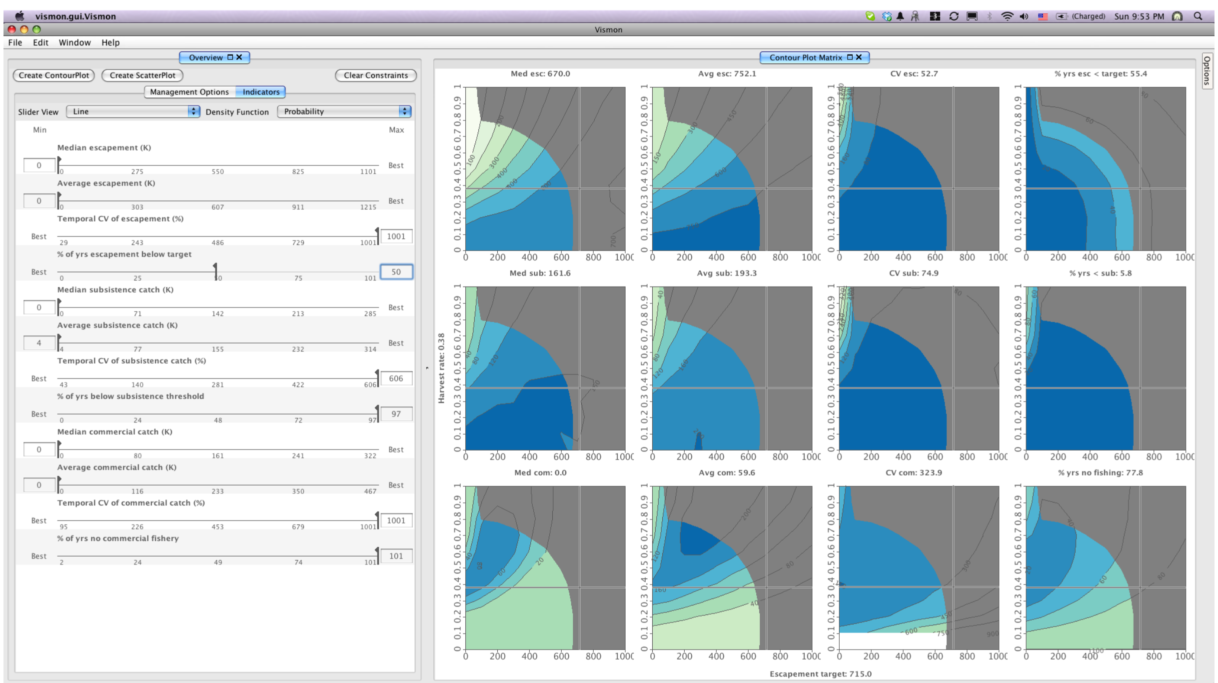

Figure 6: After setting a constraint, the active region has an irregular shape.

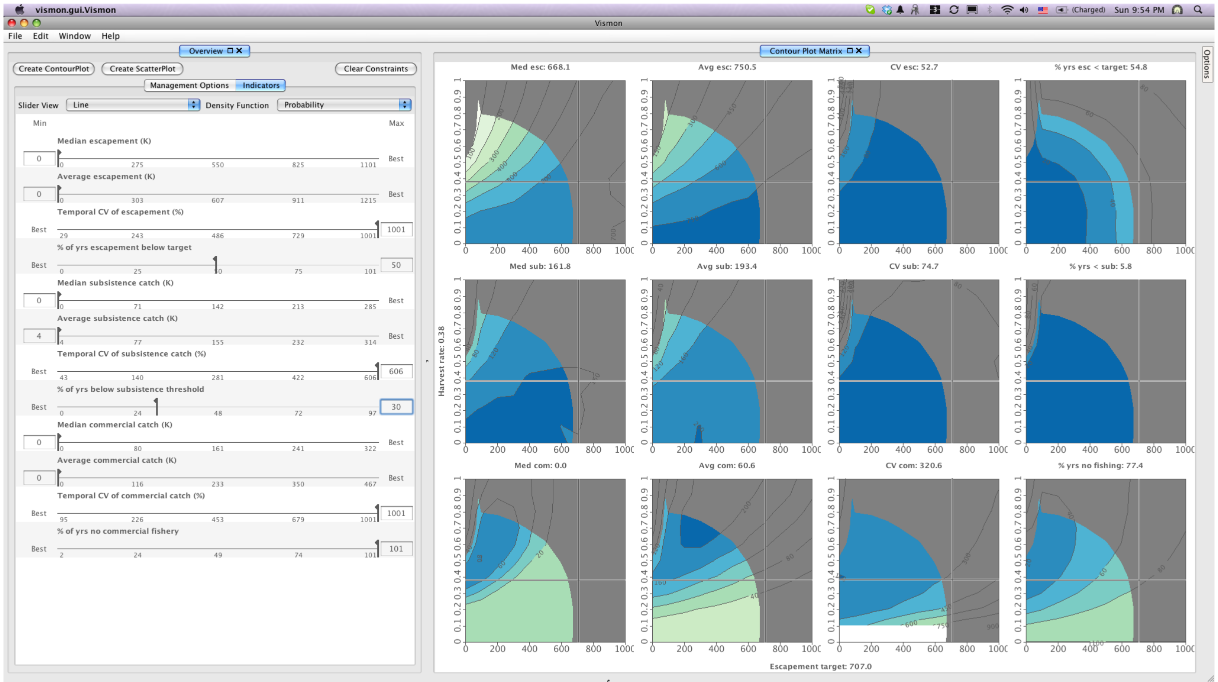

Figure 7: A second constraint shrinks the region more.

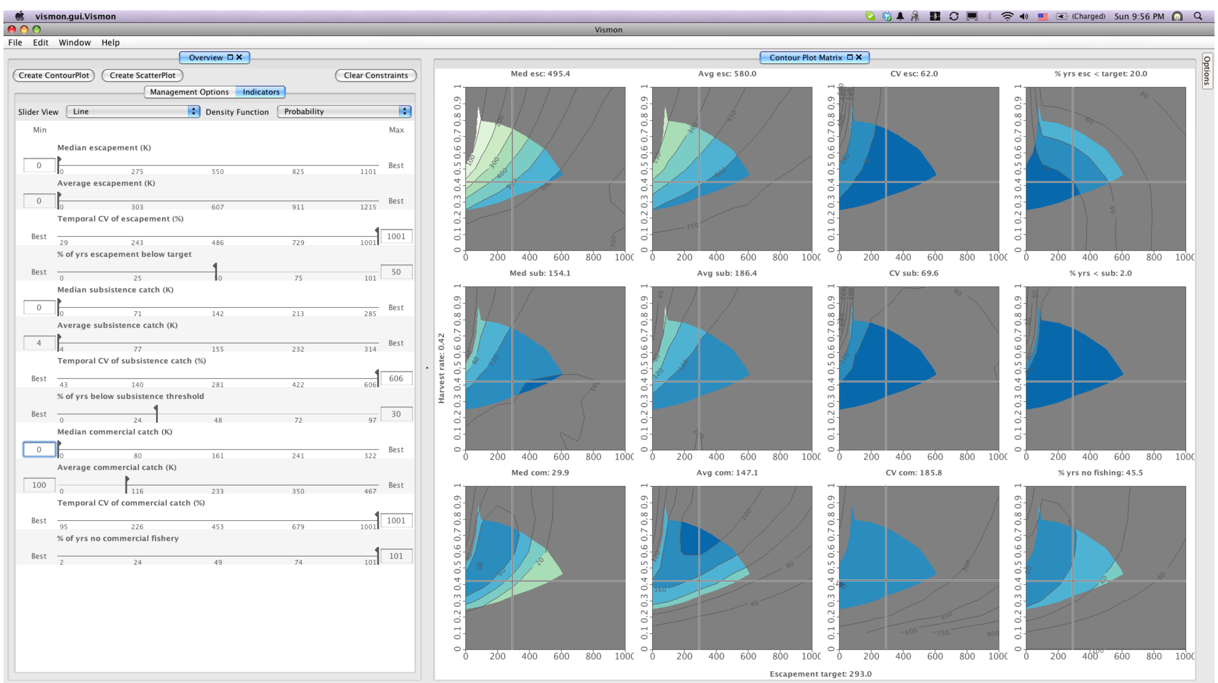

Figure 8: The third constraint shrinks the active region again.

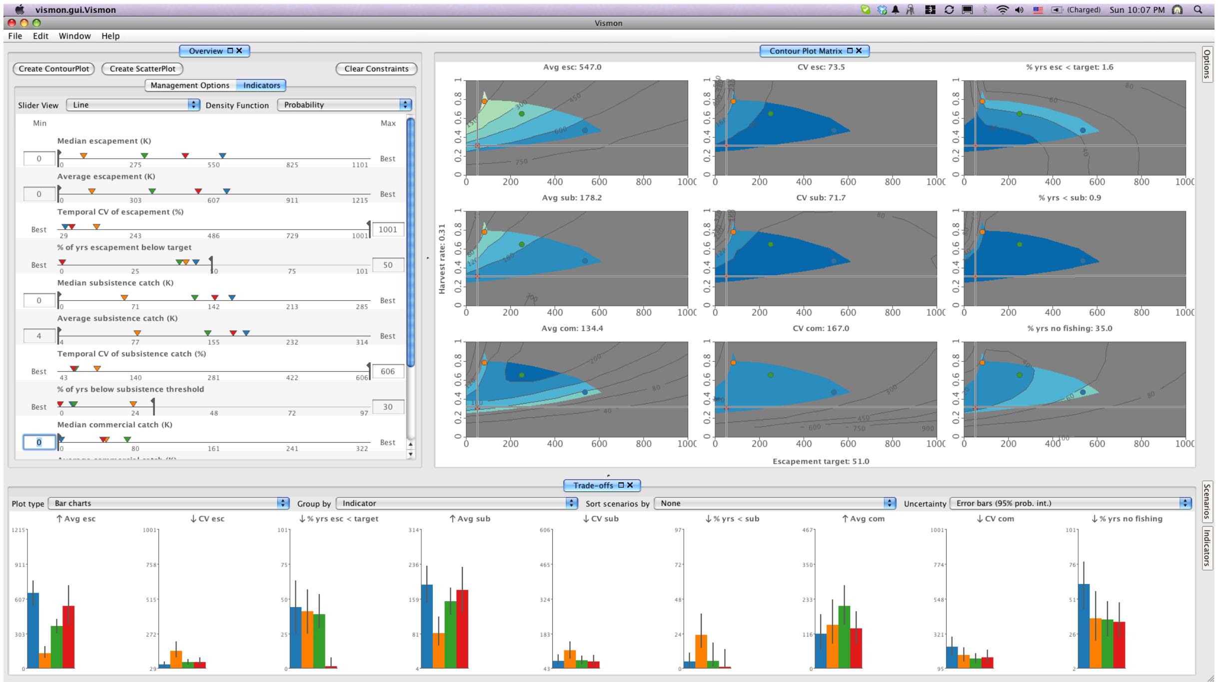

Figure 9: Clicking to select scenarios triggers the display of the Trade-offs pane.

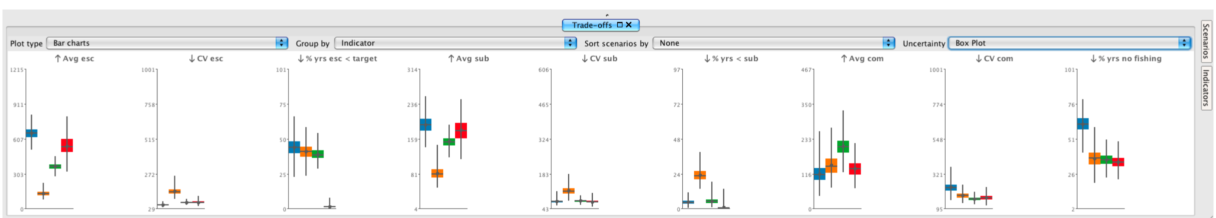

Figure 10: Switching to the box plot uncertainty display.

Figure 11: Switching to shaded distribution uncertainty and grouping by scenario.

Figure 12: Grouping by scenario, with bar charts. Labels for the bars appear on mouseover.

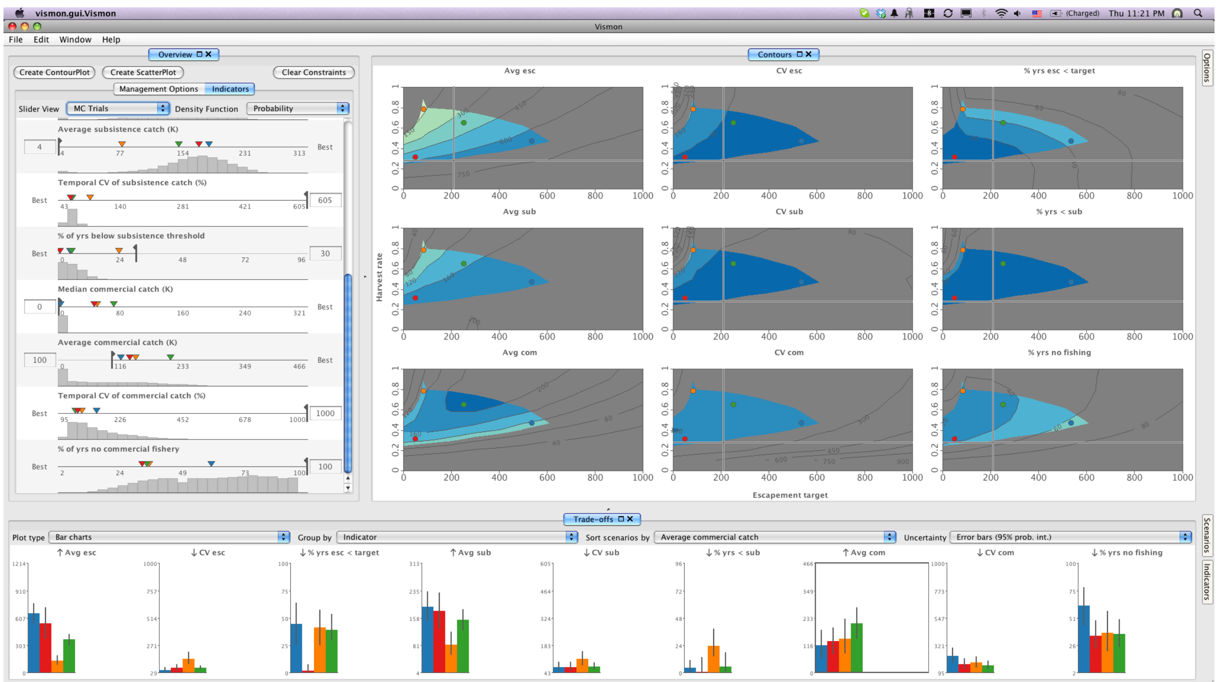

Figure 13: Returning to the indicator grouping, sorting by the `Avg com` indicator, and turning on `MC Trials` histograms.

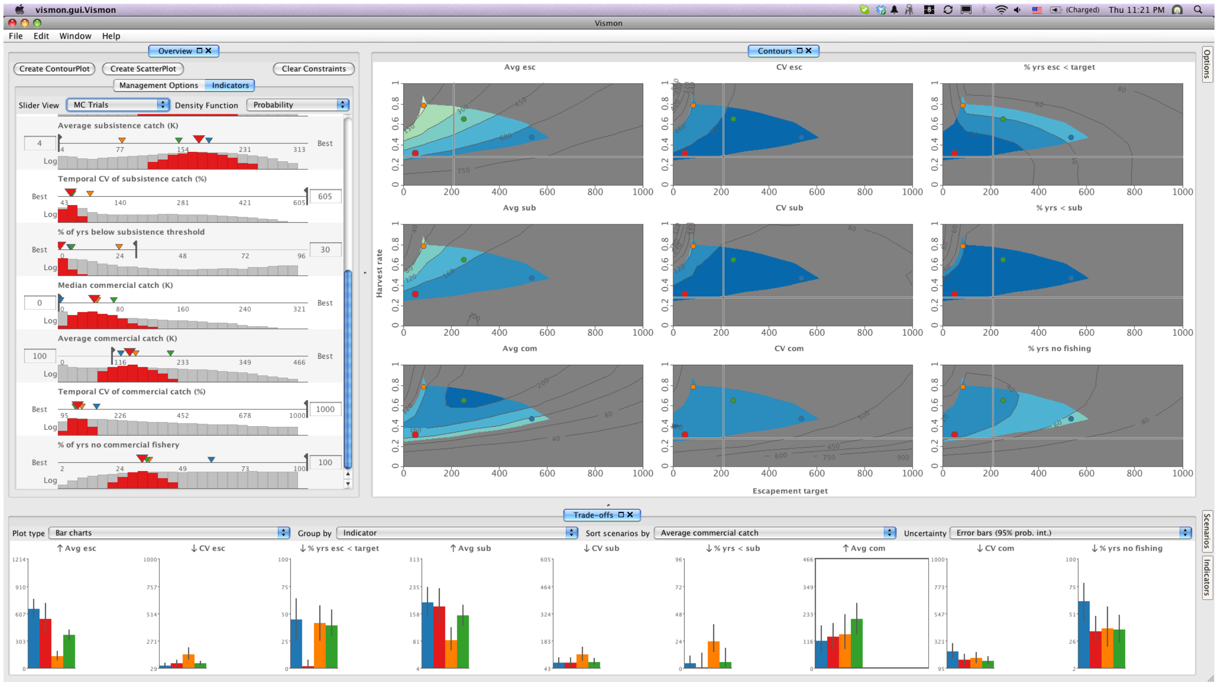

Figure 14: Choosing the red scenario shows that distribution against the rest at log scale.

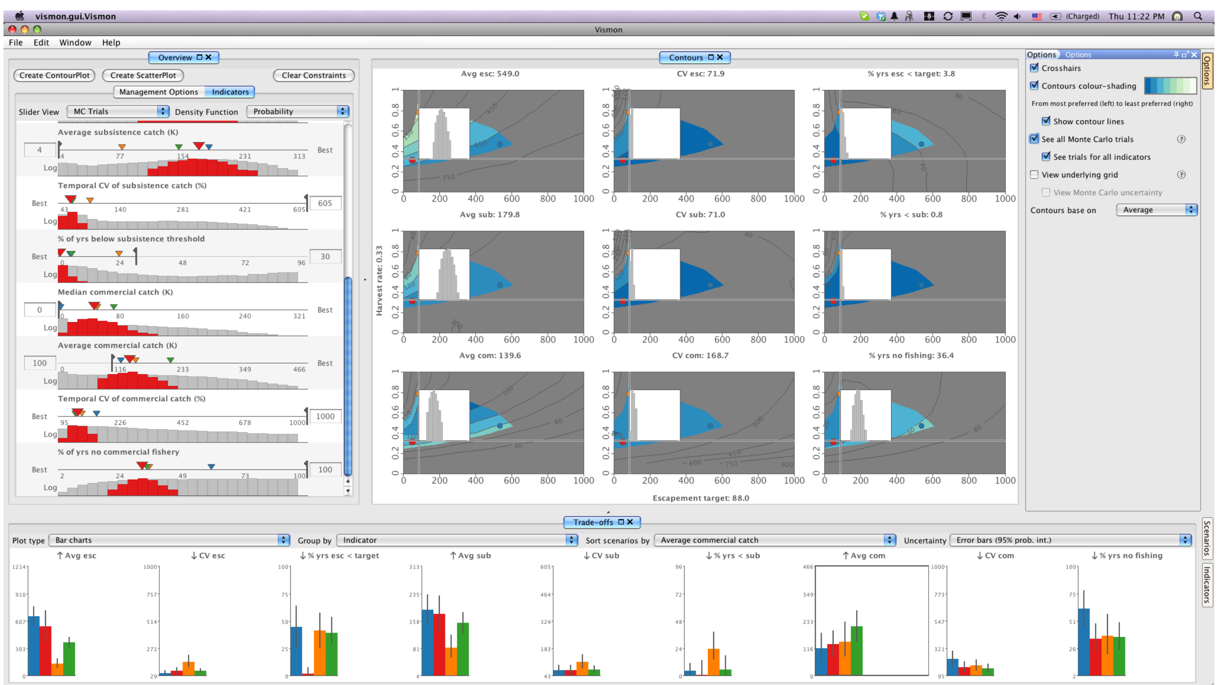

Figure 15: Turning on histograms in the Contours window.

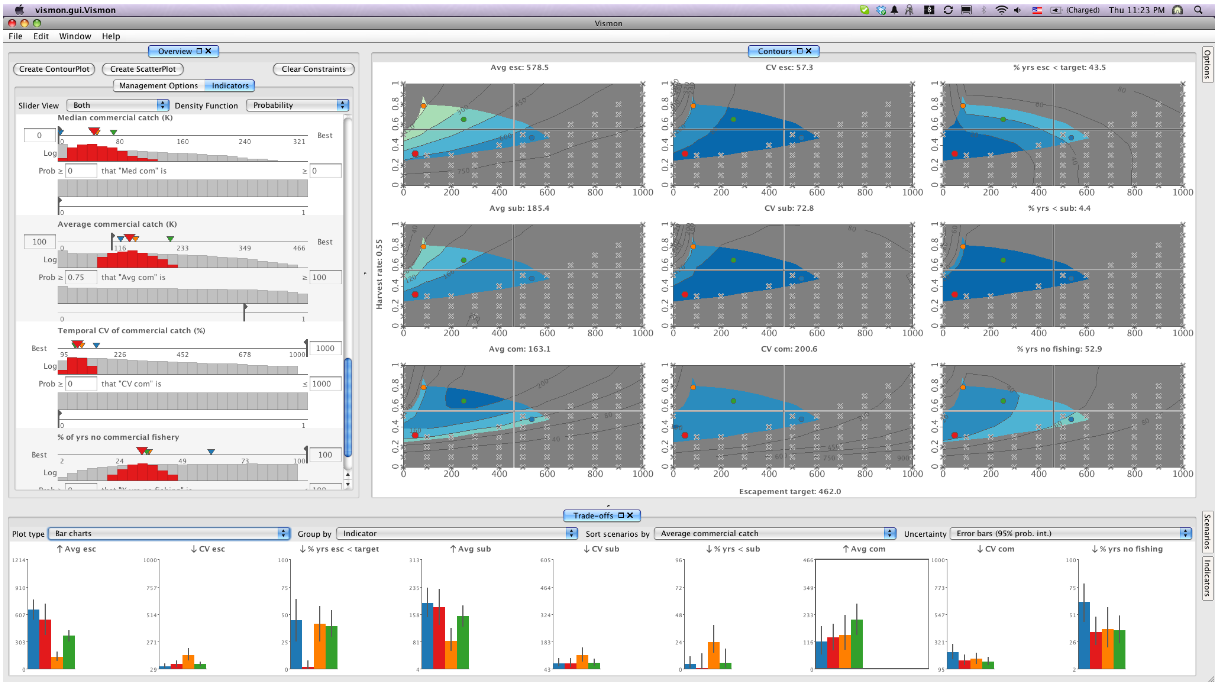

Figure 16: Using the Probabilistic Objectives slider crosses out more of the plot regions.