KTAB SMP V1.1

KAPSARC Toolkit for Behavioral Analysis (KTAB): Spatial Model of Politics GUI (SMP) User guide

Contents

This document will act as a user-guide for the KTAB: Spatial Model of Politics GUI. The guide will cover the whole process of using KTAB for the SMP model, taking the user through downloading and Installing the application, running the model, and visualizing the results. This document assumes a detailed understanding of the SMP as instantiated in KTAB. For more details, see two Discussion Papers published by King Abdullah Petroleum studies and Research Center (KAPSARC)

- *An Introduction to the KAPSARC Toolkit for Behavioral Analysis (KTAB)* , and
- *Multidimensional Bargaining using KTAB*.

Introduction

KAPSARC Toolkit for behavioral analysis (KTAB) is an open source platform for building models to allow the systematic and rigorous analysis of Collective Decision-Making Processes (CDMPs). KTAB is under continual development by KAPSARC to meet the need for widely available, state-of-the-art, supported, and open source software that facilitates the modeling and analysis of collective decision making. The software libraries and source code for KTAB are released on github[1] as open source under the MIT license.

CDMPs are those in which a group of individual actors interacts to arrive at a single decision. In both general and technical parlance, CDMPs may be termed bargaining or negotiating. To avoid confusion with legacy interpretations of these words we adopt the term CDMP, but the general meaning remains the same. The deliberations of a corporate board, the internal debates of ruling parties, the voting of electorates, and the haggling between seller and buyer at a market stall can all be viewed as CDMPs, albeit with different actors operating according to different rules. KTAB is intended to support reasonable analyses of the potential outcomes of CDMPs involving a range of modeled groups. In any group, the actors hold different values, positions, and views, which they bring to the CDMP.

KTAB is intended to be a platform that contains a number of models that can simulate CDMPs. The initial model that has been instantiated in KTAB is called the Spatial Model of Politics (SMP). The class of models represented by the SMP has a long history in the academic discipline of political science. More detail on the SMP, and references to the appropriate literature, can again be found in articles available for download on the main KTAB page. The majority of the development team's current efforts are focused on producing a GUI for the SMP. The official version with a working GUI is now available for download.

[1] http://kapsarc.github.io/KTAB/

Spatial Model of Politics (SMP)

The initial model that has been instantiated in KTAB is called the Spatial Model of Politics (SMP). The class of models represented by the SMP has a long history in the academic discipline of political science. The key distinguishing feature of the SMP is that the model assumes that positions can be arrayed along a linear, spatial dimension (usually a single line).

Fundamentally, this implies that differences in position, or advocacy, can be ordered along a position spectrum. Differences in actors' advocacy (their position) reflect the ordering and magnitude of the political difficulty in moving from one position to another. The "space" between two points on the line measures the level of difference between two positions.

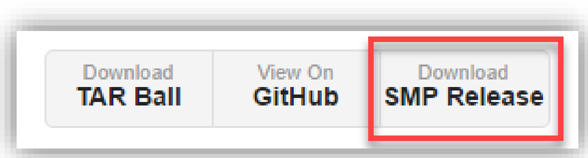
A non-spatial approach has also been implemented in KTAB. We call this the Enumerated Model of Politics (EMP).

How to Install KTAB-SMP

Download the 64-bit executable (Windows)

Installation of the SMP application on Windows is simple:

- Download and install the Visual C++ Runtime libraries for Visual Studio 2015 or more recent, from the Microsoft support official website.
- Download the latest SMP version from the KTAB official GitHub page http://kapsarc.github.io/KTAB/.
- Click the SMP Release button on the left to download a zip archive of the application compiled for 64-bit machines, along with all the required QT libraries.



- Extract the contents of the .zip archive to your PC, maintaining the directory structure in the archive.

Building from source (Linux)

Once we have configured and installed all the dependencies. the easiest way to build KTAB on Linux is to run the scripts located in the home directory of KTAB. These are clean.sh, reconfig.sh, and build.sh – they should be run in this order to rebuild all the libraries and models for KTAB.

Getting started with KTAB:SMP GUI

Changing the Working Directory

The home directory will store your default model outputs. To change your home directory, go to File > Change Home Directory

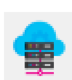
Output Log Options

You can save the model log output in different ways by indicating which logging option you prefer.

1- Default – Records the SMP model log in a timestamp-named file.
2- Custom - Records the SMP model log in a specific file / location.
3- None - Disable output logging of the SMP model run.

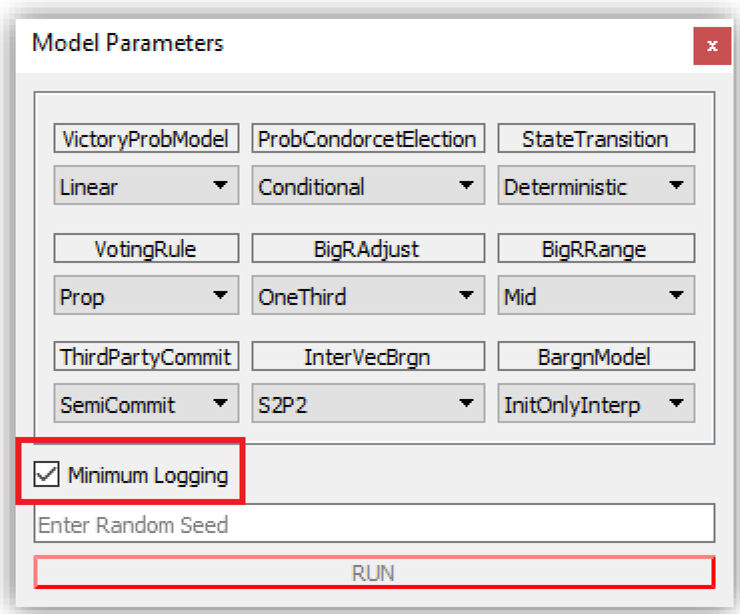
Configuring Database

You can set the database configuration to either record your model output in SQLite or PosgreSQL databases from File > Configure Database or from the database icon:



Minimum logging

Minimum logging allows the user to limit database logging. Checking this disables logging except for: model / data information. You can find the Minimum Logging checkbox within the Model Parameters panel



Note: If you need complete database output for more advanced data analysis, you should keep this option unchecked.

Model Parameters

The KTAB version of the Spatial Model of Politics is highly parameterized, with many options that can be set to model collective decision-making processes based on your understanding of how actors behave. We briefly describe the parameters that users can modify without modifying the source code. These parameters can be set either in the GUI application, or stored in an xml-format input file. As previously mentioned, the following descriptions of the model parameters assume a detailed understanding of the SMP model as instantiated in KTAB.

The default value for each model parameter is highlighted in blue in the descriptions below.

VictoryProbModel controls the rate at which the probability of a coalition supporting an option winning against the coalition proposing it increases as the strength ratios increase. This is part of the Probabilistic Condorcet Election. If Binary is used, the result is the Deterministic Condorcet Winner.		
VictoryProbModel	Linear	A 2:1 ratio gives a probability of 2/3 to the stronger coalition

	Square	A 2:1 ratio gives a probability of 4/5 to the stronger coalition
	Quartic	A 2:1 ratio gives a probability of 16/17 to the stronger coalition
	Octic	A 2:1 ratio gives a probability of 256/257 to the stronger coalition
	Binary	Any significant percentage difference gives a probability of 1 to the stronger coalition
ProbCondorcetElection controls the how the limiting distribution of the Markov process, for the Probabilistic Condorcet Election, is computed.		
ProbCondorcetElection	Conditional	PCE uses single-step conditional probabilities
	Markov Incentive	PCE uses a Markov process in which challenge probabilities are proportional to the influence of the initiating actor(s)
	Markov Uniform	PCE uses a Markov process in which challenge probabilities are uniform
StateTransition controls how the winning bargain in an actor's queue is chosen among all bargains; either by a Deterministic or Probabilistic Condorcet Election.		
StateTransition	Deterministic	The bargain which has the strongest coalition wins, even with a very small margin
	Stochastic	The probability of winning for each bargain is proportional to its relative coalition strength
VotingRule controls how the amount of influence an actor will exert between two options depends on the perceived difference in utilities.		
VotingRule	Prop	The vote is linearly proportional to the difference in utilities
	PropBin	The vote is proportional to the weighted average of Prop (80%) and Binary (20%)
	Binary	The actor exerts all influence, regardless of the difference in utilities
	PropCbc	The vote is proportional to the average of Prop and Cubic

	ASymProsp	Influence is exerted asymmetrically: It is proportional to the difference of utilities if negative (a loss in utility). It is proportional to 2/3 of the difference, if positive (a gain in utility).
	Cubic	The vote is proportional to the cubed difference in utilities
BigRAdjust controls how accurately actor i is able to estimate, relative to an anchor of its own risk attitude, the risk attitude of actor j (which is known to the model).		
BigRAdjust	OneThird	Actor i estimates actor j's risk attitude by interpolating between them, such that its estimate is closer (2/3 anchored, 1/3 adjusted) to its risk attitude
	None	Actor i judge's actor j's risk attitude as being identical to its risk attitude
	Half	Actor i estimates actor j's risk attitude by interpolating midway between its risk attitude and actor j's actual risk attitude
	TwoThirds	Actor i estimates actor j's risk attitude by interpolating between them, such that its estimate is closer (1/3 anchored, 2/3 adjusted) to actor j's risk attitude
	Full	Actor i judge's actor j's risk attitude correctly
BigRRange controls actors' risk tolerances, and hence the curvature of their utility functions.		
BigRRange	Min	Sets risk tolerances in the range [0,1] such that actors with the most probable position are perfectly risk averse (1), while actors holding the least probable position are perfectly risk tolerant (0)
	Mid	Sets risk tolerances in the range [-½,1] such that actors with the most probable position are perfectly risk averse (1), while actors holding the least probable position are somewhat risk seeking, with an aversion of -½

	Max	Sets risk tolerances in the range [-1,1] such that actors with the most probable position are perfectly risk averse (1), while actors holding the least probable position are perfectly risk seeking (-1)
ThirdPartyCommit controls how committed a third-party actor k is in a challenge between actors i and j.		
ThirdPartyCommit	NoCommit	No matter which coalition actor k joins (i or j), actor k never changes position
	SemiCommit	If the coalition actor joined by k loses, k must take the position of the winning coalition; otherwise it does not need to change position
	FullCommit	Actor k is fully committed to the coalition it joins, and must adopt the position of the winning coalition
InterVecBrgn controls how proposed positions are interpolated between the positions of actor i and j in a bargain.		
InterVecBrgn	S1P1	Proposed positions for each actor are computed as a weighted average of their current positions, where the weights are the products of salience and probability of success
	S2P2	Proposed positions for each actor are computed as a weighted average of their current positions, where the weights are the squared products of salience*probability of success
	S2PMax	Proposed positions for each actor are computed as asymmetric shifts from their current positions, which is a function of squared salience and truncated difference in probability of success
BargnModel controls from which actor's perspective the probability of success is used to interpolate bargains.		
BargnModel	InitOnlyInterp	Bargains are only computed from the initiating actor's perspective

	InitRcvrInterp	Bargains are computed from the perspective of both the initiating actor and receiving actor
	PWCompInterp	Bargains are computed as an effective power-weighted average of both actor's perspectives

The Affinity Matrix

The affinity matrix is currently unused in the model; it is a placeholder for intended future development of the algorithm.

Data Inputs

KTAB SMP uses two main data input formats: Comma Separated Value (CSV) files and Extensible Markup Language (XML) files. We assume in this tutorial that the user understands how to obtain appropriate data.

You can find CSV and XML templates in the "***KTAB_SMP\sample_data***" folder for the Windows release folder or you can find it in the official KTAB folder of the main GitHub repository under "***KTAB\examples\smp\doc\".*** These templates show how a proper CSV or XML file input is structured for a KTAB SMP input.

The Database file .db is another type of input, holding data from a previously-run model.

Actor data description

"Actor" data that represents an individual, institution or group *(Note: actor must be no longer than 25 characters)*.

"**Description**" A description of the actor *(Note: description must be no longer than 256 characters)*.

"Influence" The relative power, or political clout, for the actor.

"Position" The stated position an actor advocates for the specified dimension; this value must be between 0 and 100.

"Salience" The relative importance to the actor of the specified dimension; salience across all dimensions must sum to *at most* 100.

**Note: Influence, Position, and Salience are typically numeric values ranging from 0 – 100*

"**Dimensions**" Each dimension represents, on a continuous spectrum between two extremes, a decision over which actors are trying to influence each other, i.e. a CDMP. By convention, dimensions are represented as the inclusive range [0, 100].

"Scenario Name" Scenario, dataset or project name.

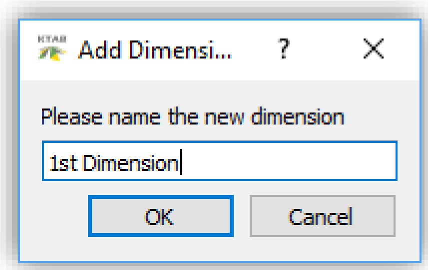
"Scenario Description" A description of the Scenario.

Create new actor data

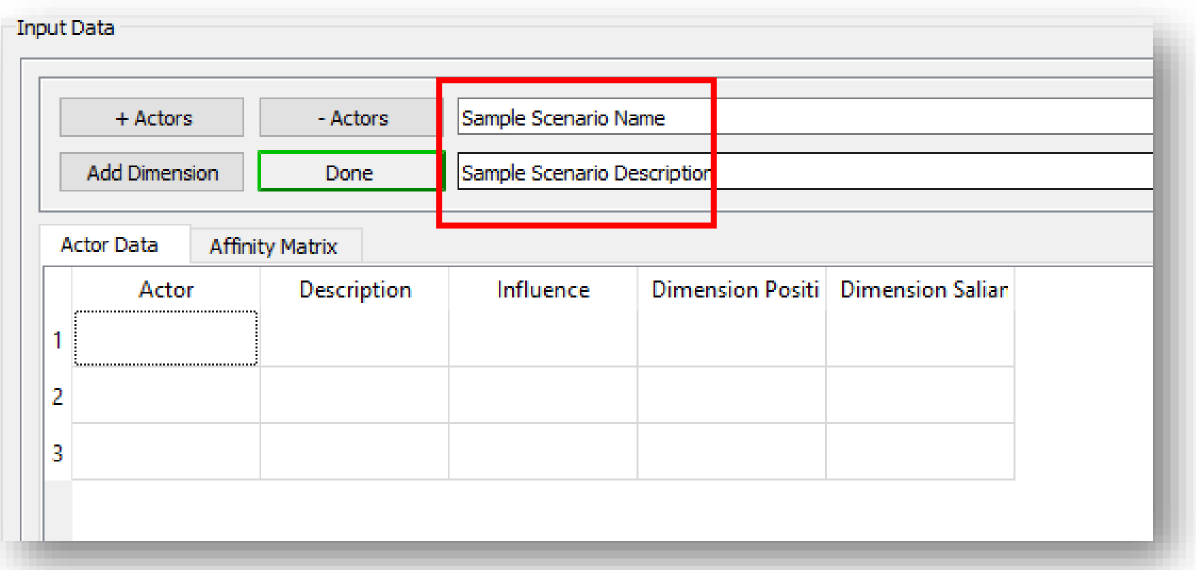
1- To start creating new actor data you can either;

- Go to *File* > *Create New Actor Data*
- Click on the icon  from the tool bar.
- Or use the Keyboard shortcut "*Ctrl+N*"

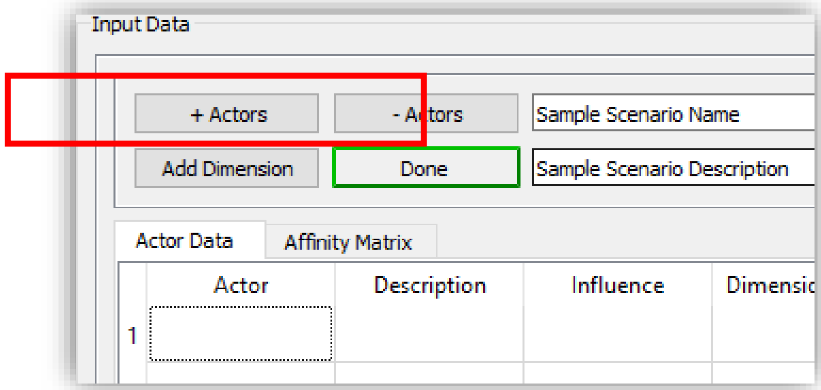
2 – Insert the number of dimensions for your new data-set



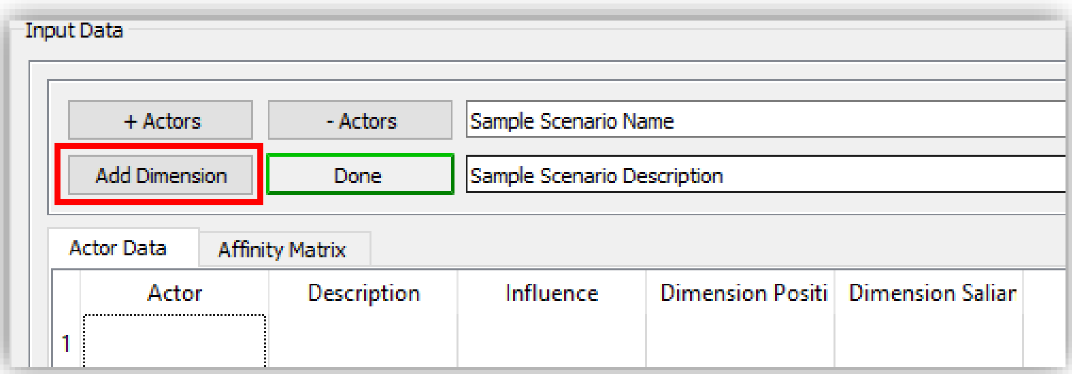
3 – To start typing in your data, you will need to set the Scenario Name and Description



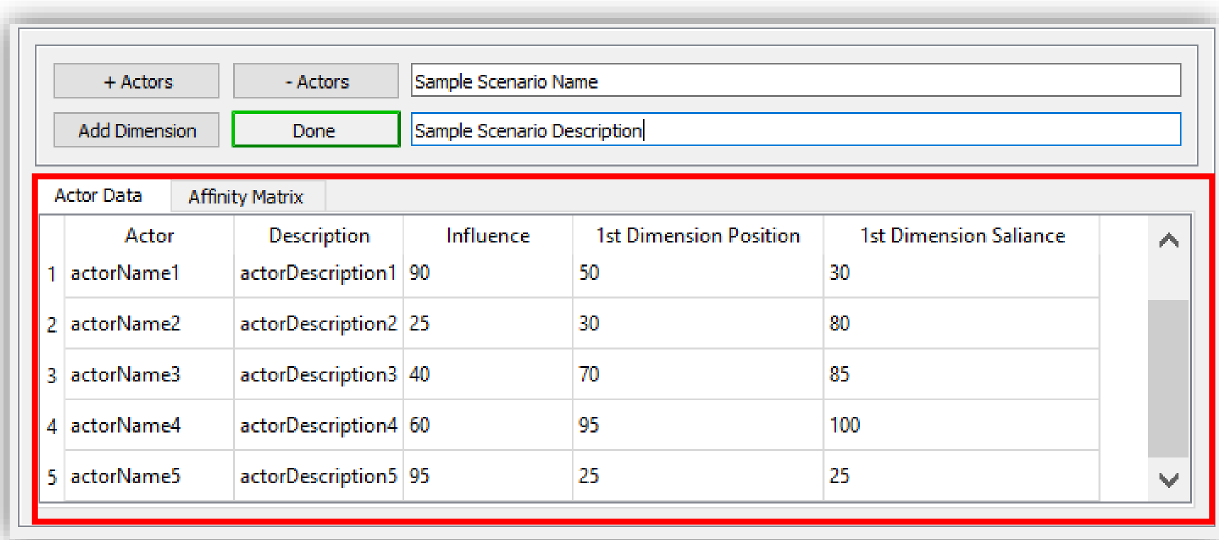
4- You can increase or decrease the number of actors using the buttons +/- buttons.



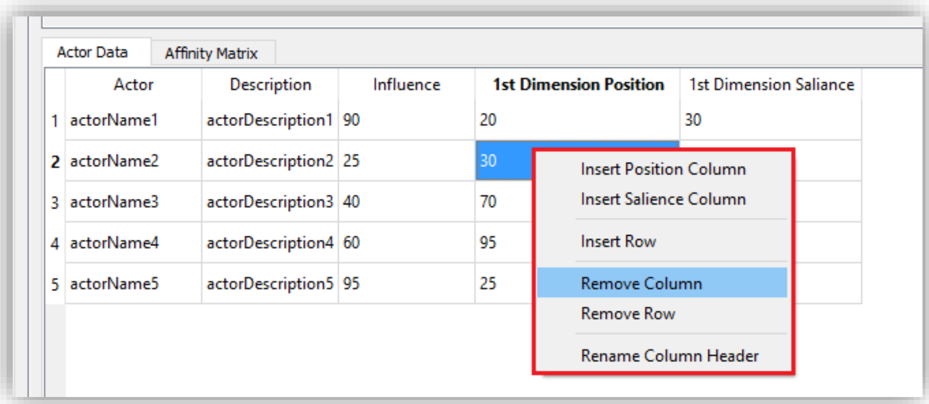
You can also click on the “Add Dimension” button to add more dimensions.



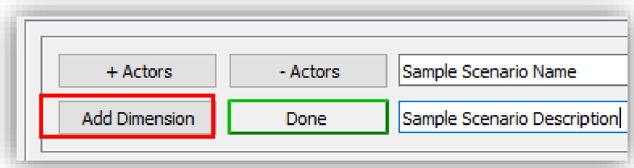
5- Type in your data in the table



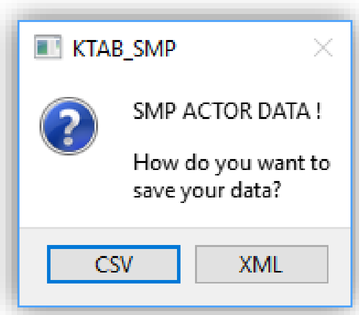
6- You can also Add/Delete dimensions, insert new actors and rename column headers by right clicking on a column from the table.



8 – Once you are done creating your new data-set, click on “Done” to save the data-set on your computer.



You can either save it in the CSV or XML file format.



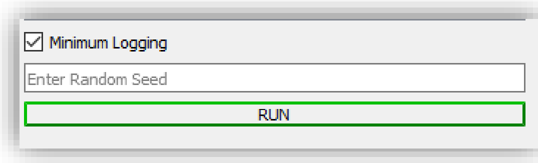
1- Data input using a CSV file

- Go to “File > View/Modify Existing CSV File”
- Or Click on the icon  from the tool bar.
- Or use the Keyboard shortcut “Ctrl+O”
- You can then choose the CSV file that contains the data that you wish to use.

2- Data input using an XML file

- You can do this from “File > View/Modify Existing XML File”
- Click on the icon  from the tool bar.
- Or use the Keyboard shortcut “Ctrl+X”
- You can then choose the XML file that contains the data that you wish to use.

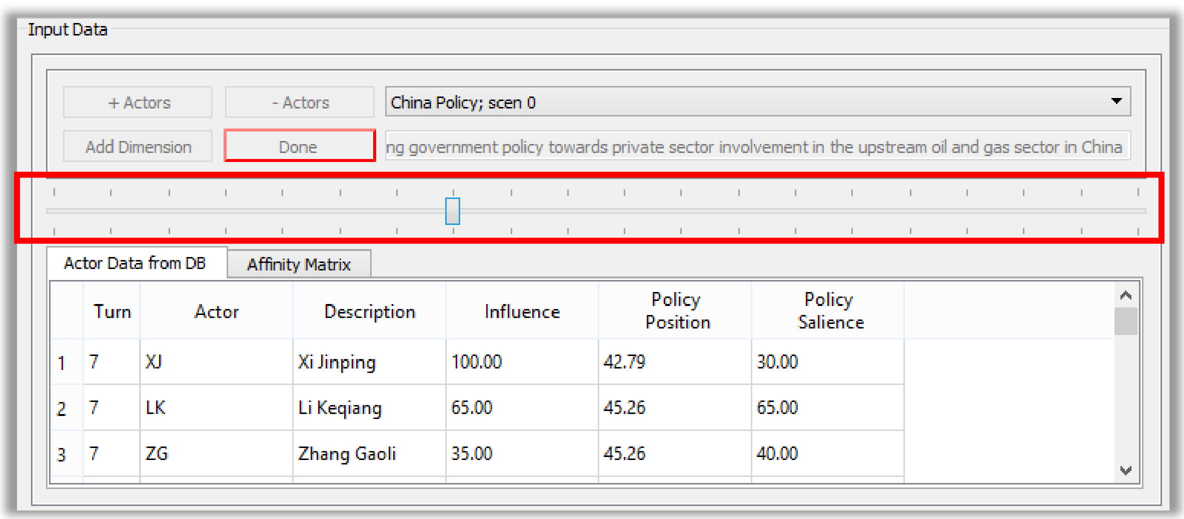
When you are done inserting your input data, the Run button will be highlighted in green for you to run the model.



Output and Results

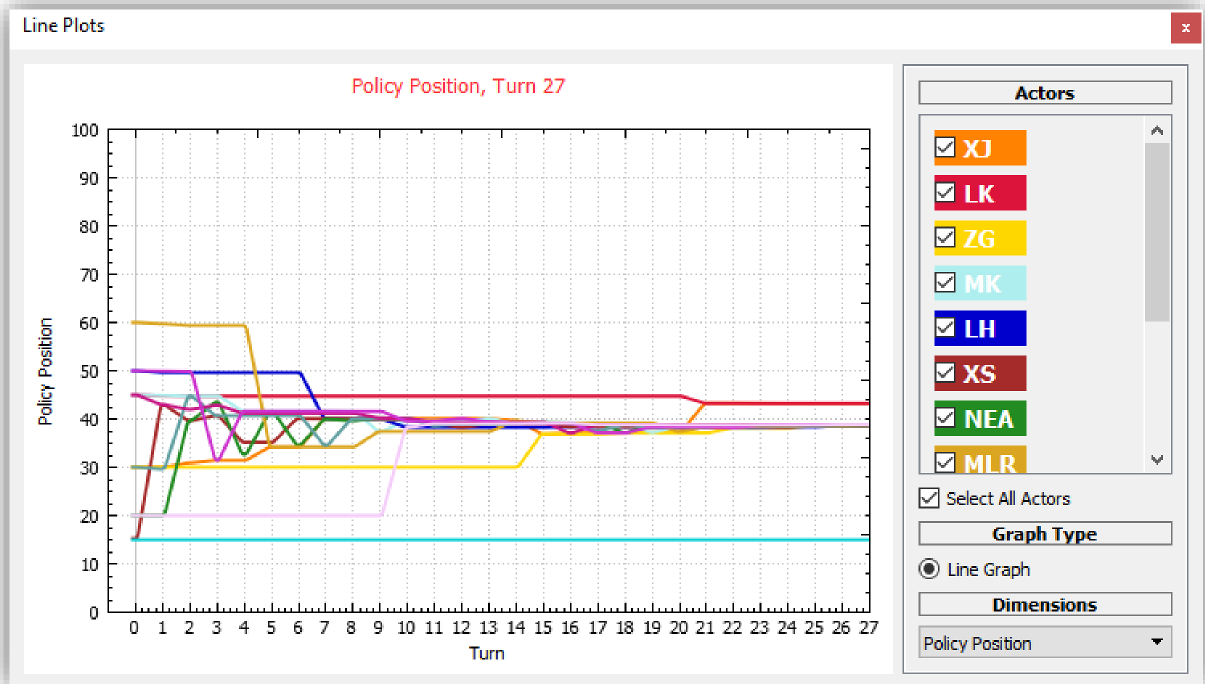
Actor Data

The Actor's Data table shows the current turn, actor's name and description, actor's influence, actor's salience and the position for the current turn. You can navigate through turns with the slider.

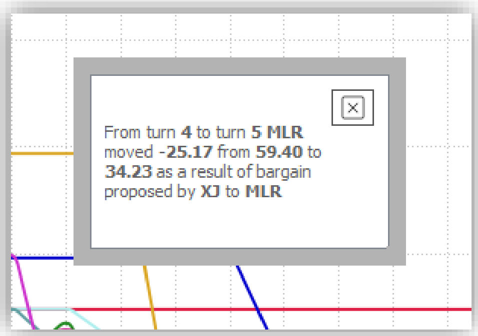


Visualizations

Positions by Turn (Line Chart)



The line chart displays the change in the actors' positions over turns. Each line corresponds to an actor, hovering over a line will trigger a tooltip showing the actor's name, description, and influence.

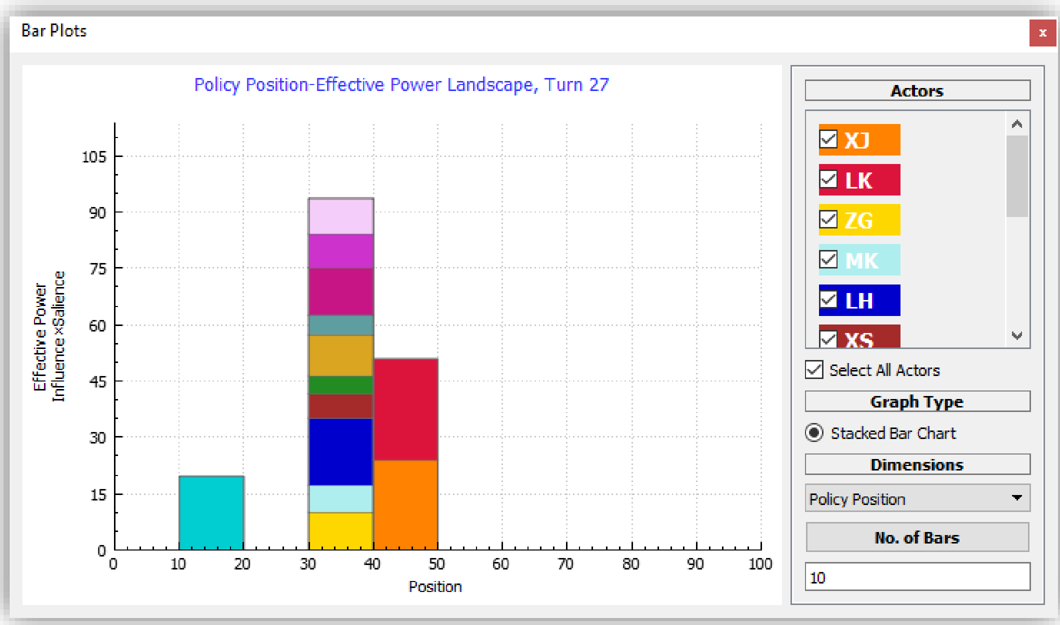


In the actor's controls on the right, you can show/hide the lines through the checkboxes for the actor. You can also switch between dimensions from the "Dimensions" section.

On the chart, you can click on the line to toggle the name of the actor and you can also zoom in/out to change the chart scale.

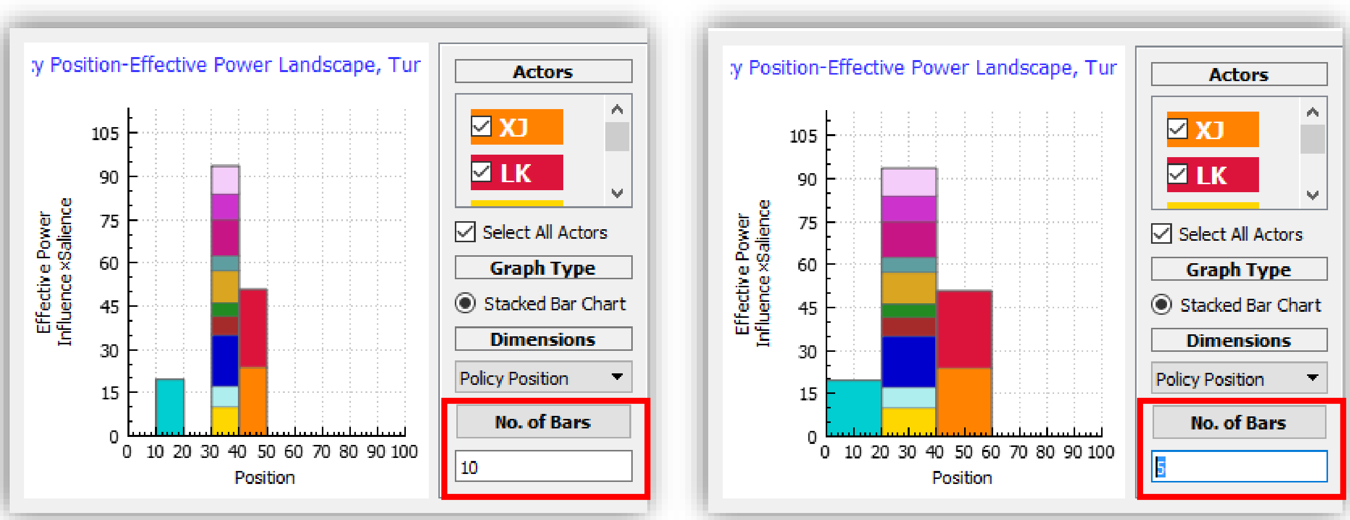
When you double click on an actor's line, changing position for a specific turn, a small dialogue box will appear explaining why the actor's position changed for that turn.

Effective Power Landscape (Stacked Bar Chart)



The bar chart displays the effective power landscape, showing the effective power and positions of actors for each turn.

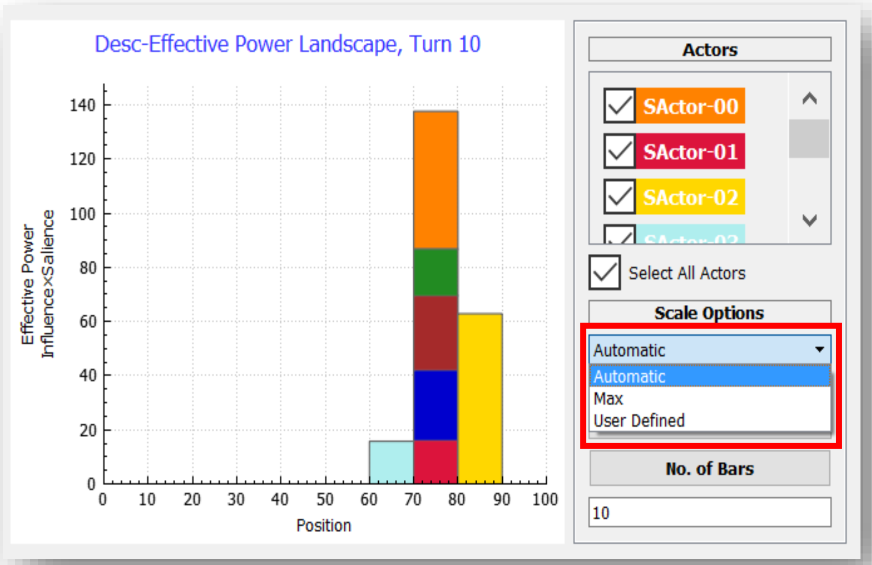
Similar to the line chart, in the actor's controls on the right, you can focus or unfocus bars through the checkboxes for the actors. You can also switch between dimensions from the "Dimensions" section. For the stacked bar chart, you can change the number of bars displayed by modifying "No. of Bars".



Bar chart scale potions

You can change the scale options to automatic, max, or user defined.

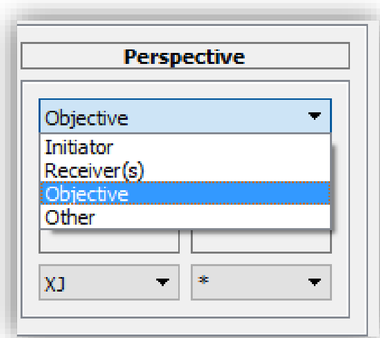
- “Automatic” is the default scaling for the bar-chart where the height of the y-axis is responsive and automatic in each turn.
- The “Max” option locks the y-axis on a fixed max scale.
- The “User-defined” option enables the user to define the max value for the y-axis.



Quad Map

You can enable the Quad Map view from *View > QuadMap*. The Quad Map plots the change in utility to both the initiator (y-axis) and receiver (x-axis) expected as a result of a challenge from a specified initiator to at least one other actor. There are four ways to compute the perspective from which the utility changes are computed:

- Initiator: All computations performed from the perspective of the initiator.
- Receiver(s): All computations performed from the perspective of each receiver selected for plotting.
- Objective: The utilities for the initiator are computed from their perspective, and the utilities to each receiver are computed from their own perspective.
- Other: All utilities are computed from a specified third party.



The quad map is named as such because the plane is divided into four quadrants, between which the behavior of receivers can be expected to consistently vary. In the first (green) quadrant, both the initiator and receiver can expect to gain utility from a challenge. Hence, actors in this quadrant may be expected to cooperate. The second (red) quadrant contains the cases where the initiator is expected to gain utility, while the receiver is expected to lose. There is likely to be conflict between actors in this quadrant. In the grey third quadrant, both actors would expect to lose utility from a challenge, so it's unlikely any would be made. The last (yellow) quadrant contains cases where the initiator would be expected to lose utility, but the receiver would gain. Again, it's highly unlikely any challenge would be made.

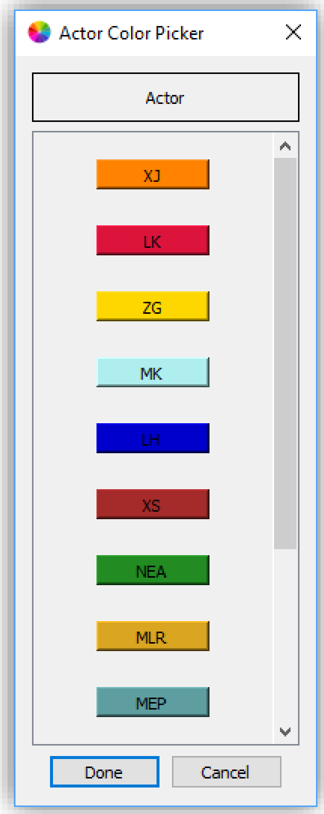
You can check the auto-scale option to change the scaling to focus on where the actors are scattered on the Quad Map.


E[ΔU] Quad Map for Actor XJ, Turn 19
0.02
0.01
0
-0.01
-0.02
-0.03
-0.04
-0.05
E[ΔU] to Initiator
-0.02
0
0.02
0.04
0.06
0.08
0.1
E[ΔU] to Receiver
Plot
Auto Scale

Changing Actor's Colors

To Change actors' colors displayed in the charts you can;

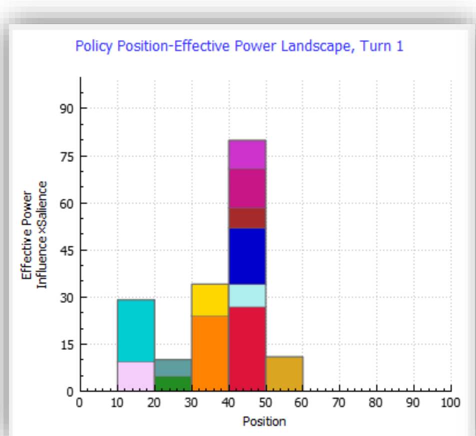
- Go to "Color Options > "Change Actors Color"
- Or Click on the icon  from the tool bar.

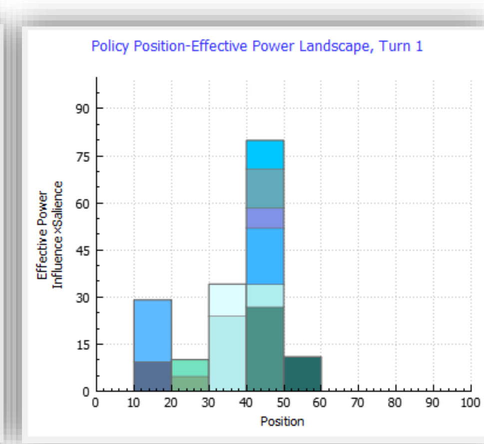


You can pick a color for each actor from the color picker widow, or you can import a color palette (which you previously designed) as a csv file from "Color Options" > "Import Actor Colors".

You can also Export a CSV color palette after modifying, so that you can use the same color map in the future, from "Color Options" > "Export Actor Colors".

Or reset the default colors for the actors. From "Color Options" > "Reset Actors Colors".





You can also Hide and View charts from the "View" option on the toolbar.