Contents

Chapter 1

User's Manual

1.1 Installation

This section describes the pre-requisites of running the application. The application has been developed using *Java SE Development Kit 7* on a *Windows X64* operating system, using *Eclipse IDE*.

It is a desktop application, which relies on internet connection for updating ontology measurements based on their URI, but can function without an internet connection using a local ontology repository, which may not be up to date.

1.1.1 Java Runtime Environment

In order o run the application on a ***Windows*** operating system, you need to have a ***Java Runtime Environment*** version 1.7 or higher installed.

1.1.2 WordNet

Dictionary application *WordNet2.1*[1] needs to be installed prior to running the application for the dictionary knowledge base access.

The path of *WordNet* installation home directory (ex: *"d:/jde/WordNet"*) needs to be set as a *System Variable* under the name *WNHOME*. This system variable name is set in the application as a static field in *ApplicationConstants* class, from where can be edited.

1.1.3 MySQL Server

The application requires communication with a running instance of *MySQL Server*[2], version 5.6 or higher. A local server connection (*jdbc:mysql://localhost:3306/*) needs to exist for the following credentials: username - *root*; password - *root*.

[1] http://wordnetcode.princeton.edu/2.1/WordNet-2.1.exe

[2] http://dev.mysql.com/downloads/mysql/

The database schema *ontologies* needs to be imported on the above server instance from the *dump* file located in project
project environment/prerequisites/ontologies_database_schema.sql.

The application database connection can be re-configured editing file
META-INF/persistence.xml.

1.1.4 File Resources

The file structure of folder ***project environment*** contains the file resources needed to run the application.

- ***prerequisites*** folder contains the database dump needed for creating and loading the database prior to running the application.
- ***ontology evaluation files*** folder contains ***initial.pwc*** file which loads the *AHP* decision problem. It also contains problem files with pre-filled pairwise comparisons with different degrees of inconsistency (***consistent1*** and ***consistent2***, ***medium_inconsistency***, ***high_inconsistency*** and ***demo*** file used as example in this chapter screen-shots), which can be imported using the AHP evaluation module GUI.
- ***local ontology repo*** folder contains the downloaded ontology files. They are used by the system as a local backup, when the corresponding online resources are not available or internet connection is disabled.
- ***AHP Ontology Evaluation System*** folder contains the runnable java application.

If one desires to alter the resources file structure, *ApplicationConstants* class constants need to be updated with the new file paths:
ONTOLOGY_LOCAL_REPOSTORY_FOLDER, *EVALUATION_FILES*, *REPORTS_FILES*.

1.2 User's manual

This section describes the main functionality of **AHP Ontology Evaluation System**.

1. Before and during running the application, the user can consist the ***MeasurementReport.pdf*** from the reports folder, in order to be familiar with available ontologies and their most recent measurements. The application opens with the domain coverage screen (Figure 1.1), where a user can evaluate the knowledge coverage of available ontologies for a given domain and preselect a subset for AHP evaluation.

2. Write a noun inside **Concept(noun)** text field and click **Get Synonyms** button to get the list of synonyms grouped by word meaning in the right pannel. If you wish to add

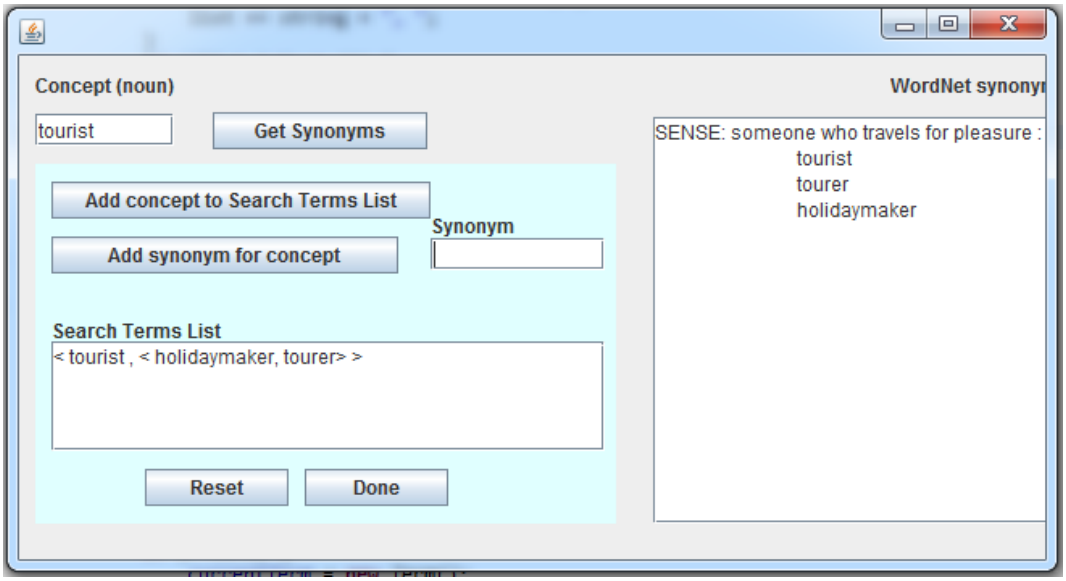

Figure 1.1: Domain Coverage Screen

current**Concept(noun)** to domain definition, press **Add Concept to Search Terms List**. If you wish to add some of the suggested synonyms to the search term, write them one by one in the **Synonym** text field and press **Add synonym to concept**. The user can reset the *Search Terms List* that defines the domain by clicking **Reset** button.

3. Add new concepts repeating the steps above. As can be seen in Figure 1.2, some concepts may have different word meanings. The synonyms corresponding to the desired sense must be selected. The user can add concepts to the *Search Terms List* without adding synonyms for them, but this decreases the chances of finding classes corresponding to that concept.

4. When you consider the *Search Terms List* is complete, press **Done**. This is the most time-consuming step of the application, you must wait until processing completes.

5. Once the domain coverage processing has completed, the dialog box shown in Figure 1.3 is shown. The user can consult the generated ***DomainCoverageReport.pdf*** from reports folder to see the values for each ontology.

6. In order to pre-select ontologies for an AHP evaluation, input the minimum domain coverage value in the dialog text field an press **Ok**. Default value *0* selects all ontologies for further step. By pressing *Cancel*, the user can proceed with a new domain coverage evaluation.

7. Clicking **Ok** with a valid input loads the AHP evaluation screen, presented in Figure 1.4.**Stimuli** panel present the children of the selected element in **Criteria** panel. **Judgments** section presents the matrix of pairwise comparisons corresponding to the selected Criterion, that needs to be completed. To the left are the inconsistency measurements. The sub-criteria that need to be compared can be also visualized in **Decision Aid->Graph View** and **Equalizer View**.

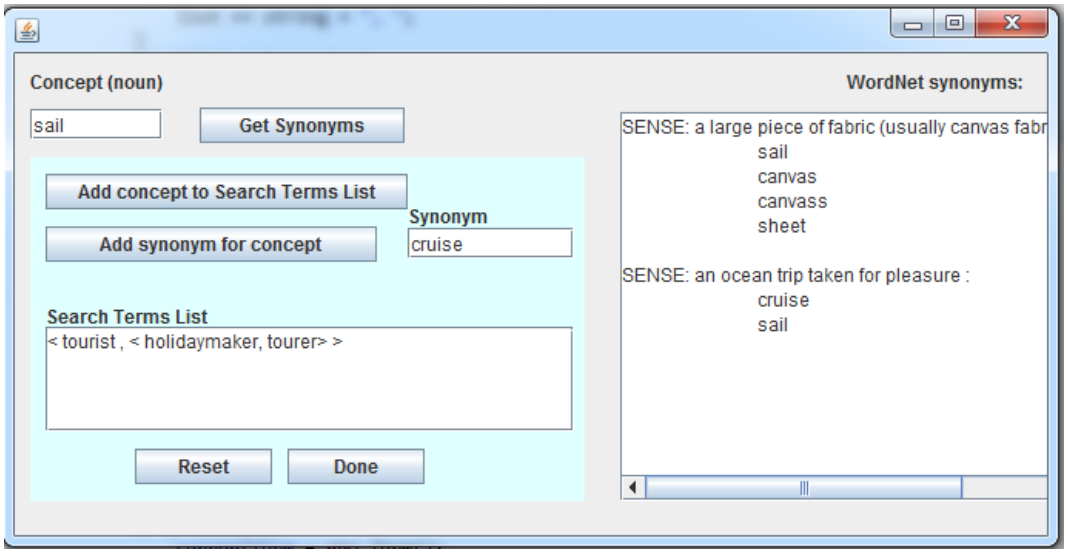

Figure 1.2: Domain Coverage Screen

The inconsistency measurements are *Consistency Ratio (CR)*, *Consistency Measure (CM)* and *Congruence (Θ)* for *cardinal inconsistency*; *The Number of Three-way Cycles(L)* and *Dissonance (Ψ)* for *ordinal inconsistency*.

8. The pairwise comparison preference judgments can be input also in the **Equalizer View** (Figure 1.5), which suggests the relation between direct and indirect judgments graphically. Direct judgments are represented as larger circles, while indirect judgment are smaller circles. The segment between criteria is a two-directional preference axis, the middle being *preference equivalence.* A perfectly consistent matrix has circles concentric for all pairs. A *latent violation* is visible when a direct judgment and an indirect one have opposite directions on the axis, being highlighted in yellow. In this example, three intransitive judgment three-way-cycles have occurred from inconsistent input (L=3).

9. Inconsistent judgments can be identified, evaluated and corrected also in the **Judgments** panel (Figure 1.6). Button **Dissonance** reveals the *Dissonance* and *Congruence* of each individual judgment. Button **Triad for CM** highlights in blue the most ordinally inconsistent judgment. Figure 1.6) highlights in red the third set of intransitive judgments (L3) listed in the **Equalizer View**.

10. When all pairwise comparisons have been provided, the user can press the **Evaluate!** button to obtain the final evaluation values for pre-selected alternatives. The user can consult the generated ***AHPEvaluationResultReport.pdf*** to obtain a detailed documentation about the current evaluation process.

11. The final values for ontology alternatives (given by id) is shown in **Problem** tab. The correspondence between alternatives and evaluation values is color-coded. This tab also contains the problem description and use guidelines.

12. By selecting a non-leaf element in the **Criteria** panel, the elicited weights of

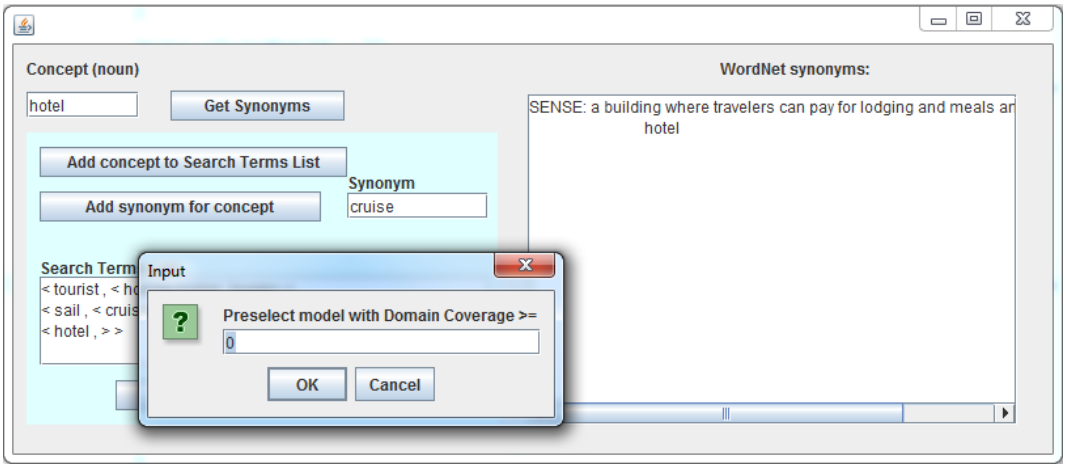

Figure 1.3: Domain Coverage Screen

its sub-criteria can be seen in **Vectors** tab. **Gnatt View** displays the values in a manner similar to Gnatt diagrams, suggesting the relation between value magnitudes visually (Figure 1.8). **Numeric Vaues** tab (Figure 1.9) displays also the elicitation accuracy measurements: *Total Deviation form Direct Judgments (TD)*, *Total Deviation form Indirect Judgments (TD2)* and *Number of Priority Violations (NV)*.

13. By selecting a leaf element in the **Criteria** panel, the elicited weights of alternatives for the corresponding Criterion can be seen in **Vectors** tab. The example from Figures 1.10 and 1.11 displays the alternative weights for *Average Number of Sub-classes* atomic criterion. As in the previous step, the relation between weight values and **Stimuli** is color-coded.

14. The user can **export** the problem to a *.PWC* file or **import** a new problem for evaluation.

15. The application is exited by clicking the exit button in the upper right corner of the window. Unless exported, the decision problem is not saved. Before running the program again, the user is advised to save the generated reports in a different location or with a different name, as they will be overwritten.

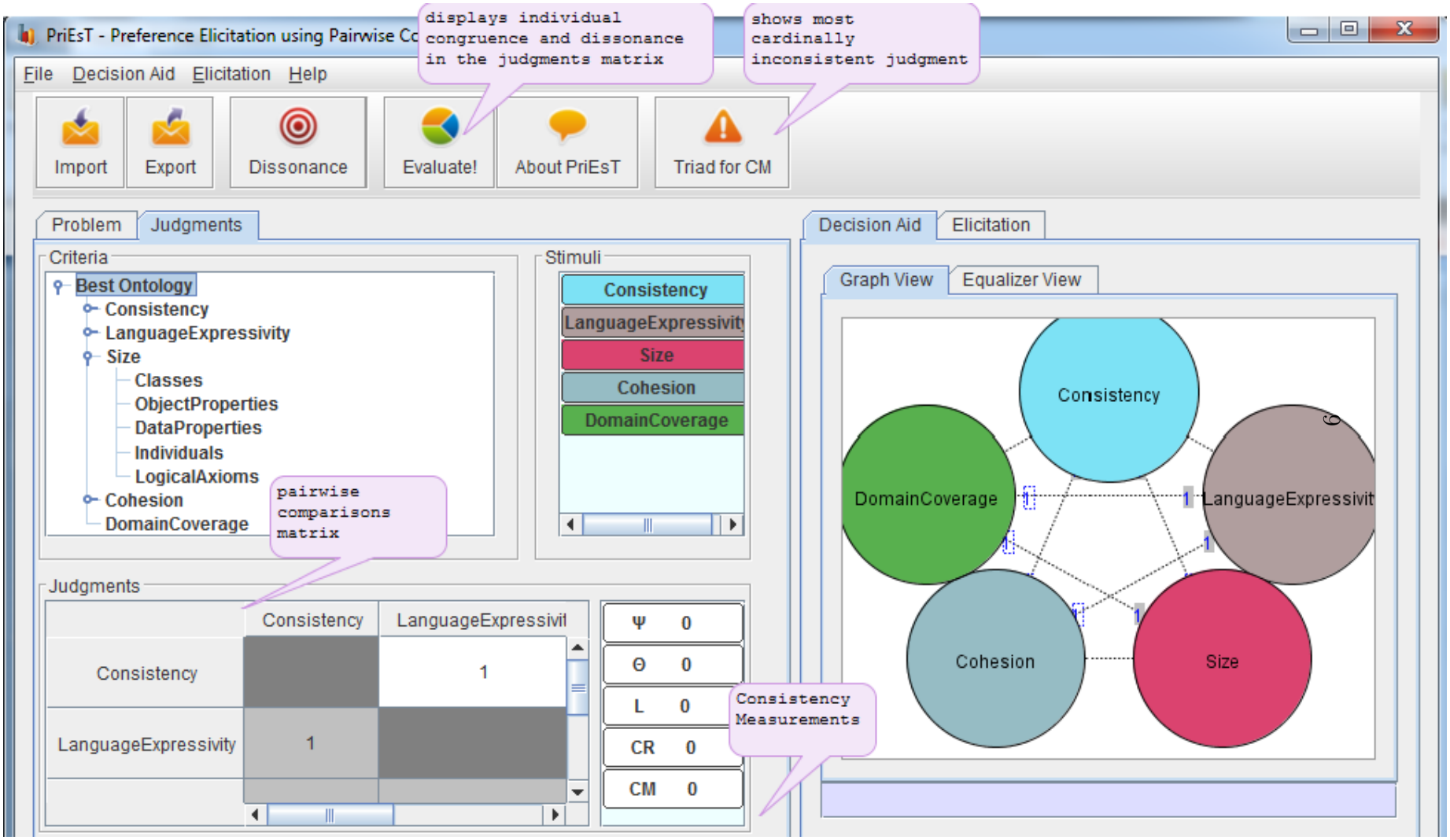

Figure 1.4: AHP Evaluation Screen

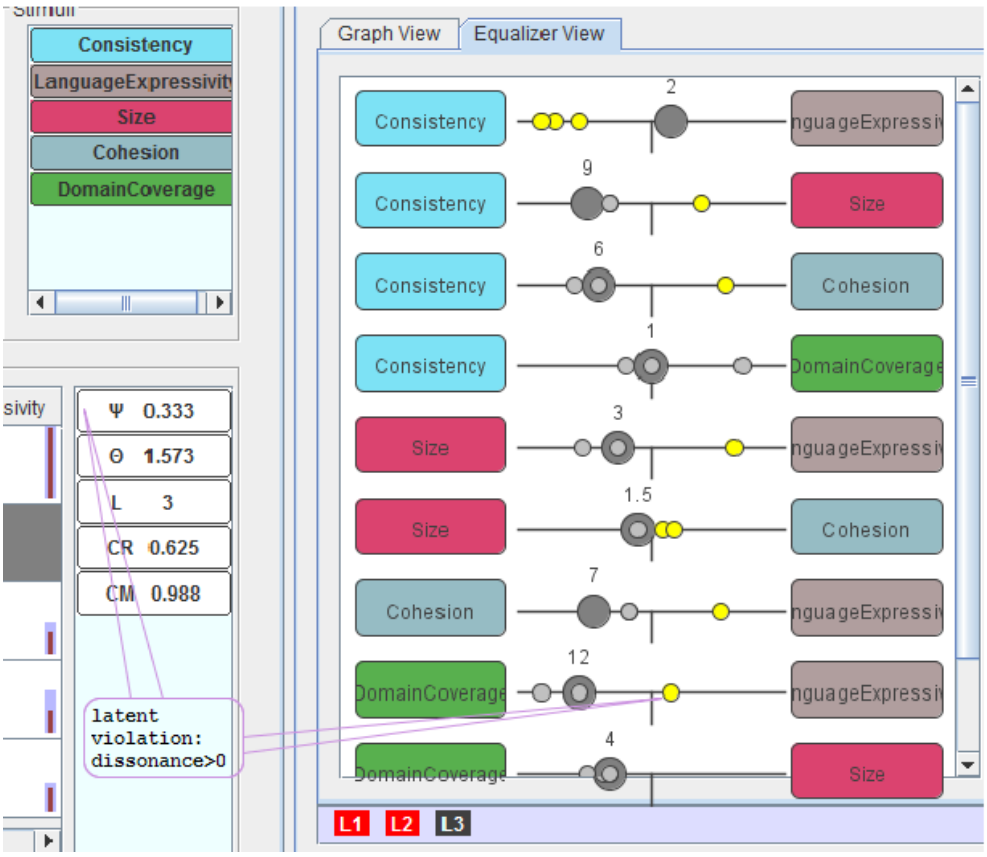

Figure 1.5: AHP Evaluation Screen: Equalizer View

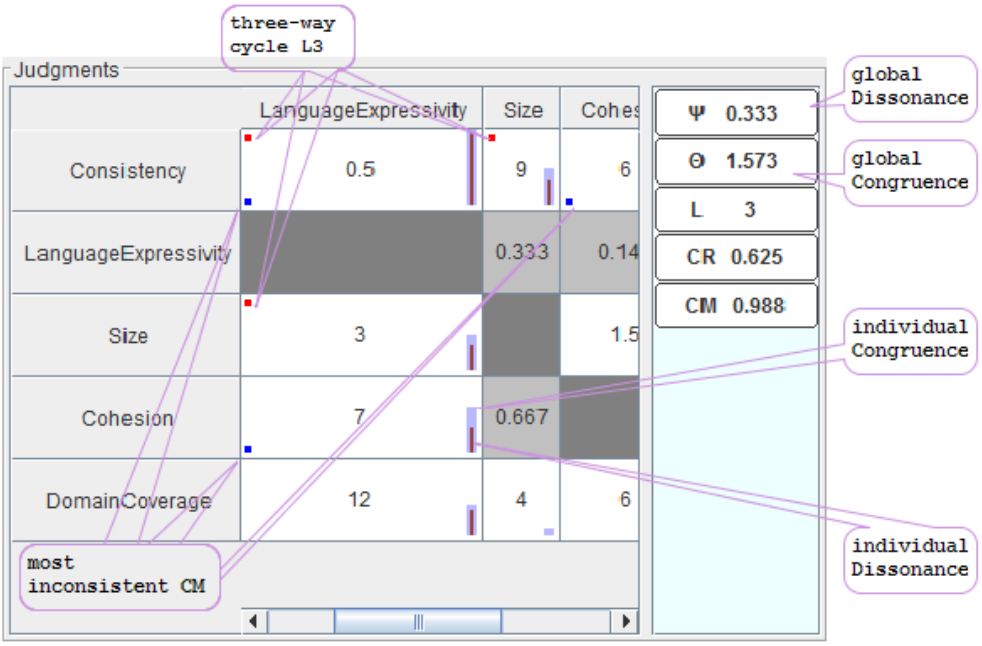

Figure 1.6: AHP Evaluation Screen: Judgments View

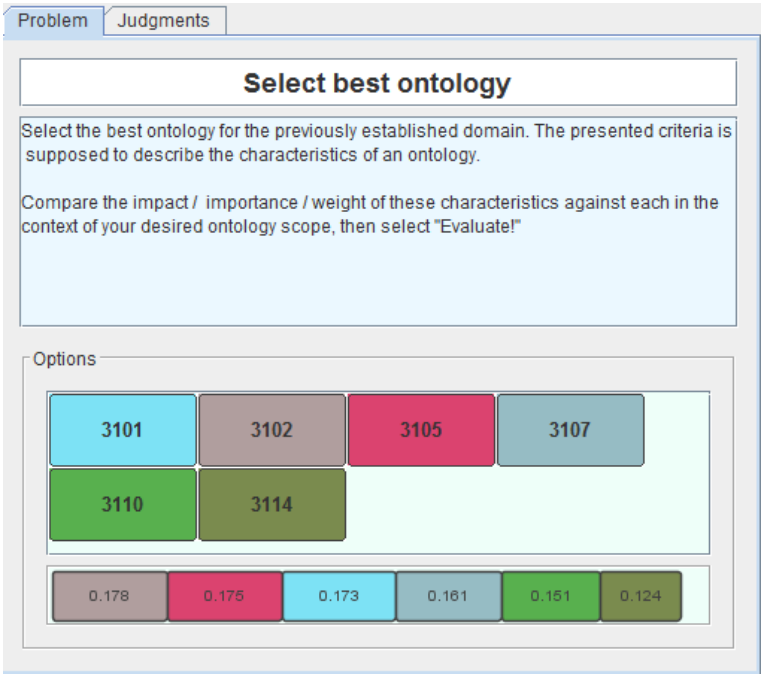

Figure 1.7: AHP Evaluation Screen: Problem View

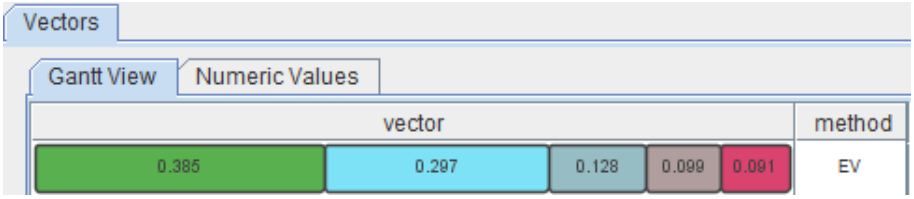

Figure 1.8: AHP Evaluation Screen: Criterion Weights

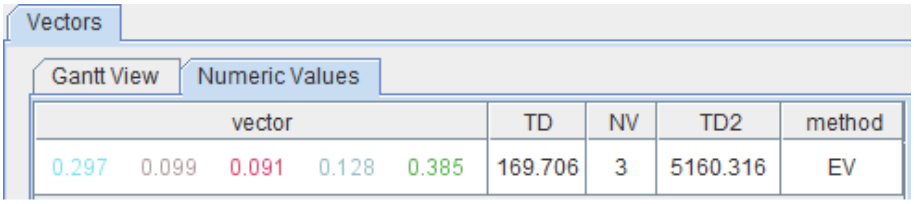

Figure 1.9: AHP Evaluation Screen: Criterion Weights

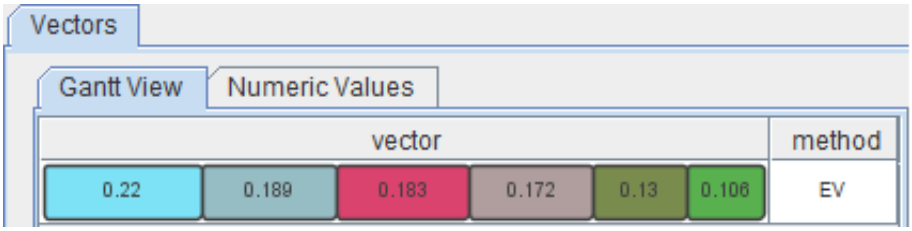

Figure 1.10: AHP Evaluation Screen: Alternative Weights

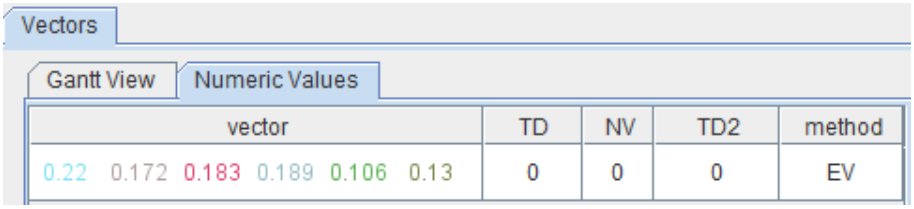

Figure 1.11: AHP Evaluation Screen: Alternative Weights