Visual Reflection: Language, Action and Feedback

Luís M. Carriço[†‡] *lmc@di.fc.ul.pt*

Pedro M. Antunes[†] *paa@digitais.ist.utl.pt*

Nuno M. Guimarães[‡] *nmg@di.fc.ul.pt*

[†] *Instituto Superior Técnico, Technical University of Lisbon, Av. Rovisco Pais, 1049-001 Lisboa, Portugal*

[‡] *Faculdade de Ciências, University of Lisbon, Campo Grande, Bloco C5, 1700 Lisboa, Portugal*

Abstract

This paper addresses the direct manipulation of cognitive maps. These maps are visual languages aiming at the representation of the conceptual structures of reasoning. Constraints are imposed to the articulation of the visual elements of the language in order to convey its syntax and semantics. Those constraints are often defined with several degrees of uncertainty as they result from the expression of beliefs. Therefore, the tools that offer a direct manipulation approach for editing cognitive maps must provide the necessary visual elements to build the maps and support feedback dialects that convey different resistance levels.

This paper proposes a set of visual language elements for cognitive maps and its manipulation and a model that identifies relevant dependency properties on those elements (the parent's model). It identifies the actors of a manipulation and proposes a couple of metaphors (magnet's and membrane's) in order to communicate resistance levels. The corresponding feedback dialects, using the properties of the parent' model and operating on particular actors of the manipulation, are presented.

1. Introduction

Cognitive maps are visual languages used to represent and understand thought [14, 15]. They aim at the classification of concepts and identification of relations that reflect the structure and dynamics of the reasoning processes. Cognitive maps include several levels or formalization and detail and focus primarily on the exploration and comprehension of cognition. Formal representations are, thus, just one facet of cognitive mapping. Often, concepts, relations and their properties result from the expression of contradictory beliefs (or uncertain knowledge) of several experts.

This orientation to explore and specify knowledge while profiting from the advantages of a visual representation [14] lead to the application of cognitive maps to several domains. In artificial intelligence, these graphical representations are used as semantic networks [22, 20] to formally represent knowledge and natural language. In design [21], they were applied to capture design rationale [10, 12] and in writing to support collaboration [23, 16]. In management, cognitive maps were used to study causal thought [11, 15, 3] and classify variables [19] in strategic decision making. Also, in organizational diagnosis [18], they are used to understand crucial problems.

The most common representation of cognitive maps uses graph-based diagrams. Nodes (vertices) symbolize concepts and arcs (edges) designate relations. Visual attributes (e.g. labels, shape or line thickness) reflect the properties and types of concepts and relations. More elaborated languages for cognitive mapping use structuring techniques in order to facilitate users' perception. For instance, collapsing groups of concepts and relations into a more general concept simplifies the resulting diagrams. Hierarchical regions can be defined on the map space to reflect categories.

The constraints to the above visual elements convey the syntax and semantics of the underlying conceptual model. Contrary to formal languages and simple syntactic constraints, the semantics and uncertainty of the represented knowledge generates particular constraint forms. For example, in a cognitive map a constraint could only state that a causal relation (arc) between symptoms (nodes) probably should not be drawn (with a certain degree of evidence).

Direct manipulation is adequate to facilitate the analysis and construction of cognitive maps. This is stressed by the visual nature of these languages, by the exploratory nature of the construction and revision process and by the targeted users (e.g. management consultants and psychologists). The manipulation problems that it raises are similar to the ones found in drag-and-drop operations available in common window managers. However, the dynamics and uncertainty of the constraints introduces further obstacles and new challenges, particularly in the definition of semantic feedback to reflect those constraints. Simple icon changes, that forbid or allow a manipulation, are insufficient.

This paper addresses the dynamics of visual languages, in the context of cognitive mapping. First, it presents an articulated set of basic components, which can be combined to assemble specific cognitive mapping languages. Then, it describes its active counterpart, including direct manipulation operations and specific interaction objects. In the following section, the article proposes an object model that defines the set of attributes that can be adjusted in order to reflect specific constraints and strengths towards manipulation. It also identifies the physical and logical actors involved. Their shape, properties and relative positioning can be arranged in coherent feedback dialects to show multiple visual responses to manipulation. Two sample dialects are presented, based on two distinct metaphors (magnet's and membrane's) and built upon the proposed model. In the final sections, we present some related work, focusing on techniques and tools for manipulation of graph-based diagrams in general and cognitive maps in particular. Finally, conclusions are drawn and future work is considered.

2. Language

Figure 1 shows a joint cognitive map of two negotiators during a negotiation process. Although simplified, it

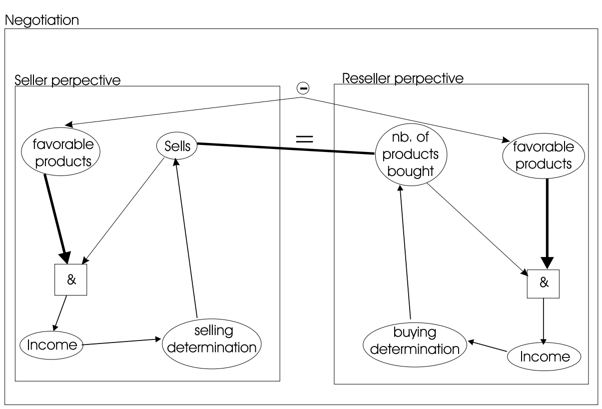


Figure 1. A cognitive map of a negotiation.

presents the essential elements found in elaborated cognitive maps. The basic elements are: **Concepts**, **Associations**, **Operators** and **Contexts**.

Concepts are nodes that symbolize the variables involved in the reasoning process. In figure 1 they are shown as ellipses containing a textual description of the variable.

If required, different shapes can be used to differentiate the concept type (e.g. goal, plan, belief or fact). Other visual attributes may reflect other properties of those variables. For example, line thickness can be used to denote the importance of a concept and line breaking the evidence of a symptom on a diagnosis (e.g. medical or organizacional [7]).

Associations are arcs and symbolize relations between concepts. Relations may have distinct types (e.g. part of, cause, equivalence) and characteristics (e.g. certainty, weight). The type and properties of the concepts may determine, encourage or discourage the existence of certain types of associations (e.g., concepts of completely different types will hardly be equivalent). As arcs, their ending positions depend on the positions and shape of concepts they link. A label, textual or iconic, may be attached to the arc to represent the relation type. Arcs can also be directed depending on the relation.

In figure 1, the unlabeled arcs represent positive causality, i.e. an increase on the origin concept causes an increase on the destination one (e.g. the bigger the seller's income the bigger his determination to sell). Two arcs have labels: an equal and a minus sign. The first one represents equivalence and is not directed, whereas the second one means negative causality and in this case is bi-directional (the products more favorable to the seller are less favorable to the buyer and vice-versa). Finally, the thickness denotes weight. In the case shown, favorableness of products is more relevant in the determination of income that the number of traded products.

Operators are nodes that combine associations. They usually represent binary logical operations like `AND` and `exclusive OR`. Unary operations are defined in the relation type, e.g. positive and negative causal relations. Operators depend on the associations they join, i.e. the association type indicates what operators may or may not apply. In figure 1 two AND operators are used to convey that income increases only if there is trade and profit.

Contexts are regions that group concepts. Contexts may simply provide additional meaning to the concepts they enclose or define constraints that determine the type of the concepts or some of their properties. When constraints are defined, they are imposed on the concepts. For instance, if a context denotes a concept type, then all the concepts visible therein must be of that type. Associations may cross contexts. However, contexts may impose constraints directly on the existence of associations. For example, a context may recommend a maximum number of associations for all their concepts.

In the figure, three contexts are represented as rectangles: the global negotiation context, the seller and reseller perspectives. In this case, the inner contexts refine the concepts meaning. For example, the income variable becomes the seller's income in the left context and the reseller's income on the right one.

Contexts can be visually overlapped (see [8] for further details). In the overlapping area, the constraints associated to each context are merged according to one of the following operations: subtraction or conjunction. With subtraction, only the constraints of the top most context are imposed. With conjunction, the constraints of both contexts are added (contradictory constraints are disabled).

Concepts, Associations and Operators are **dependent** objects. The objects they depend on will be called **dominants**. Dependent objects are those that have properties that are determined by attributes or constraints defined by the dominant object. In the case of concepts, dominants will always be contexts. With associations, dominants will be concepts (at least two) and contexts. Operators have associations as dominants.

The above-mentioned elements correspond to the representation perspective of the visual language i.e. elements symbolize some characteristic of the represented reasoning process. Particular concept categories (e.g. issue, position, argument), imposing constraints on specific associations (e.g. part-of, suggests, refutes) with determined operators (e.g. and, or) and framed within special contexts (e.g. immediate level, narrative level) provide the means for the specification of rather complex reasoning processes. This static perspective of the visual language will be designated the language level (see figure 2).

3. Action

To reach specifications, users engage on a construction and revision process, where the language components are created, modified and deleted, often in an exploratory way. That interaction with the language elements provides a richer form of expression and reflects the language action-oriented nature. This dynamic perspective of the visual language will be designated the action level (see figure 2).

At this level and for a direct manipulation style, the main interaction approach will be ”drag and drop”. It is used to adjust the layout of the diagrams (e.g. moving nodes within its context, or reshaping arcs without changing endpoints) and to alter the meaning of existing objects. For example, to reclassify a concept, the concept must be picked and dragged from one existing concept to another. Even the

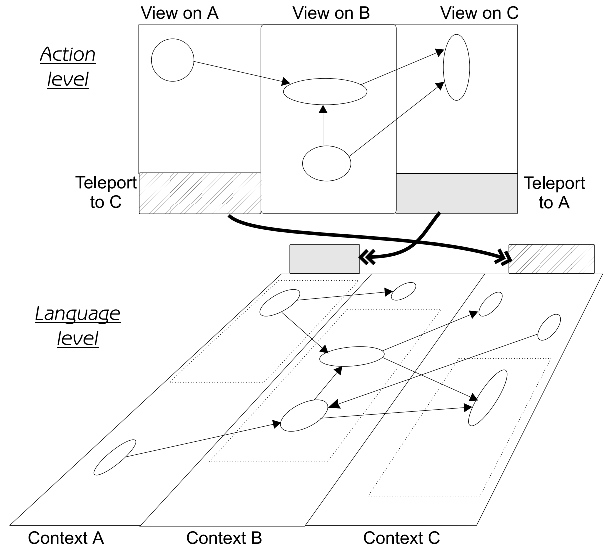


Figure 2. Elements at the Action and Language levels.

creation of new elements can be considered a drop operation (for concepts) or a sequence of drop drag and drop for elements that relate (associations and operators) or group (context) more than one object. To create an association, the association is dropped in the initial concept, dragged and dropped into the ending concept. To define an operation the involved associations must also be gathered. More complex alternatives, e.g. for path definition, are always decomposable into simple movements.

During the creation and modification of cognitive maps, manipulations with other meanings are required as well. For example, merging or decomposing problem descriptions (concepts) is pointed as a common action in organizational diagnosis methodologies that use cognitive mapping techniques [18]. Drag and drop operations can also be used with these meanings and others. In addition, it can be done with one or several objects, thus facilitating repetitive tasks. For instance, defining several causal associations (dropped in a multiple selection of concepts) with a common consequence (concept) is also usual.

These interactions are applied to the objects of the action level. Having adopted a direct manipulation style, these objects include straight representations of the language elements. Therefore, concepts, associations, operators and contexts exist at both levels. However, a language element may have more that one representation at the action level. In particular, two perspectives of contexts where introduced (see figure 2): **Teleports** and **Context Views**. These objects are facilitators for the interaction with elements of cognitive maps, in particular those defining visual contexts.

Teleports are regions that represent channels to other contexts (usually non-adjacent ones). They do not contain concepts. Instead, they communicate with a target context. Concepts dragged therein will enter the target context. Two teleports are shown in the previous figure. The one on the left leads concepts to context C (on the right) and the one on the right communicates with concept A.

Context Views are regions that represent windows over target contexts (a detailed description can be seen elsewhere [9]). The functionality they provide is similar to that of current toolkit views (e.g. scrolling) with two basic extensions. They change the shape of arcs, so that they flow continuously between views and they optionally hide arcs that end or begin in non-visible nodes. Context views are articulated in order to determine the shape and visibility of the arcs. An Integrated Multi View (IMV) object manages a set of context views.

The diagram in figure 2 shows three context views. The dotted rectangles on contexts refer to the areas visible through the corresponding views. If the rectangles were cut from their contexts and aligned, the resulting picture would be very confusing. Instead, in the IMV arcs run smoothly and irrelevant information is removed.

4. Feedback

In order to define rich feedback dialects, the actors involved in a drag and drop manipulation must be identified. They are the lexical terms of the dialects. Their visual properties and their relative articulation can then be assembled into the dialect phrases, able to convey to the user the constraints on the manipulation of the action level elements. On these elements, a common set of properties that reflect the different strengths and preferences of the constraints must also be defined. These properties provide the framework to construct the syntax of the dialect phases. They are described next as part the parent's model. Then the actors are presented and two sample dialects are described.

4.1. Parent's model

This model particularly defines the properties of the action level elements relative to their dependents. It is inspired on parents' typical behavior towards their own children (dependents that want to become independent) and to strangers (those willing to become dependent).

Dominant objects define a **attraction field** delimited by two boundaries (see figure 3). One boundary is inside the object and usually coincides with the object's own outline.

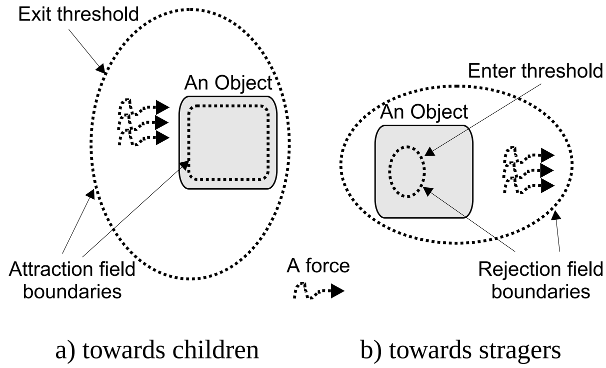


Figure 3. The parents' model.

The other stays outside the object. Manipulation of dependent objects willing to take them out of the dominants' influence feel an **attraction force** as long as they stay inside the attraction field. The **exit-threshold** on the outer boundary defines the point where the attraction force ceases. The attraction field boundaries do not necessarily have a shape similar to the object, or a common center. They may have distorted attraction fields with a range of influence depending on the direction (for example, to define preferential exiting directions).

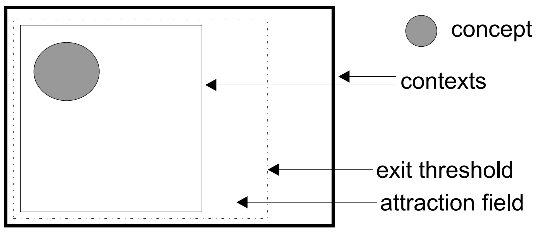
Dominant objects also define a **rejection force** and a **rejection field** (see right side of figure 3). Rejection is applied to non-dependent objects that are manipulated with the intention make them dependent. Resistance to these manipulations will be felt as soon as the manipulation actors enter the rejection field and will cease as soon as the **enter-threshold** is reached. The boundary where dependents enter the rejection field may or may not coincide with the object's outline. The rejection field boundaries may have shapes and centers different from the host object. The enter-threshold boundary can be adjusted after the corresponding field is entered. This is particularly useful for small objects where the entering boundary coincides with the object's outline. In this case, the enter-threshold can be set to the position that is further away to the field entering point.

The model is common to all the elements of the action level, including the two context perspectives: teleports and context views. They will represent the constraints of the target contexts in their own field so that manipulation feels similarly. Naturally, field shapes and forces do not need to be the same. They will be adjusted to the particular size and shape of the teleports or view.

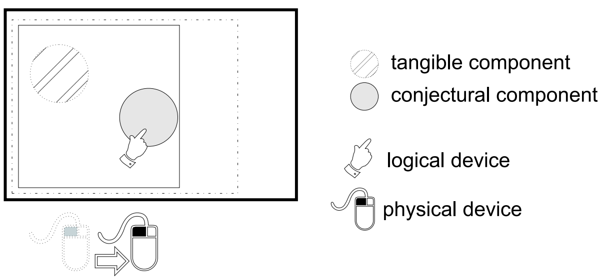
Attraction and rejection forces and fields depend on the dominant object, on the type of the manipulated object and on the manipulation meaning (e.g. copy, move and merge). They reflect the language syntactic and semantic constraints. For example, in an organizational diagnosis a problem (concepts) was identified and classified by an expert group as technical one (context). The certainty of the classification is high. A large and strong attraction field should be defined by that context to a manipulation that tries to reclassify the problem. The same reasoning can be made about the rejection field. For instance, problems referring to machinery will hardly be acceptable in a context containing social issues.

4.2. Feedback actors

The actors involved in a generic drag and drop manipulation are: the original dominant (or dominants) of the object, the destination dominant, the manipulated object and the instrument of manipulation (see figure 4). The manipulated



a) before dragging.



b) after dragging.

Figure 4. Actors.

object can be separated into **tangible** and **conjectural** components, once the manipulation starts. The tangible component refers to an already existing object, while the conjectural one refers to a prospective occurrence (which may be viable or not at the end of the manipulation). These components may reflect different manipulation states, according to the constraints on the conjectural component (e.g. ”is not at all possible as it is”, ”it will be something similar if manipulation stops now” or ”it will be exactly this”) or the manipulation meaning (e.g. move, copy, create).

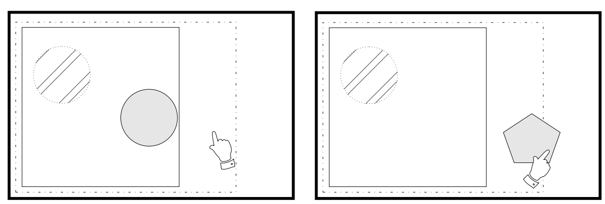
The instrument of manipulation is further classified into **physical** and **logical** components. The physical device is the mouse, the keyboard, a voice command set, etc. The logical device coincides with the usual cursors associated to pointing devices. For other devices, such as the keyboard, a logical representation is created. A strength and direction are associated to the logical device.

Feedback dialects are defined with these objects and components. Their shapes visual attributes as well as their relative positioning may be changed dynamically, at each manipulation step (even for physical devices, when possible, e.g. recent joysticks that provide reaction feedback). Usually, current user interfaces only change the shape of the logical device and one of the manipulation components (that most of the times coincides with the logical device). Next, we present two metaphors that manage actors in different ways to provide richer feedback to the user. The behavior of context views is identical to their target contexts so they will be referred indistinctly as contexts.

4.3. The magnet metaphor

In the magnet’s metaphor, the manipulation instrument acts as a magnet. It is based on the relative distance between the conjectural component and the logical device and the speed of the manipulation instrument components. Once a dependent object reaches a dominant’s field the conjectural component is separated from the logical device until the corresponding threshold is reached.

The strength of the constraint imposed by the dominant object is conveyed by the size of the field (the bigger the field the harder is to complete the manipulation) and the field force. The latter determines the speed of logical device relative to the physical device. For instance, for stronger constraints bigger mouse movements result in smaller movements of the logical counterpart. Alternatively, each stroke on the key results in less movement of the corresponding logical device.



a) are you sure? b) ok! I’ll change!

Figure 5. Leaving dominant’s influence.

Figure 5 continues the previous one and shows a situation where a concept in dragged out of a context. An object is represented with different formats (circle and polygon) to convey different properties imposed by the origin and destination contexts. The tangible component of the manipulated object is dimmed, conveying that it will be there if the manipulation is canceled. The conjectural component indicates where the object will stay if the manipulation is completed. When conjectural component reaches the beginning of the attraction field of its dominant, it stops. Then, the logical device is separated from the conjectural component (see figure 5a). Separation increases until the logical device reaches the exit threshold. At that point, the conjectural component rejoins the logical device. It also changes its shape to reflect the change on the properties imposed by the entering context.

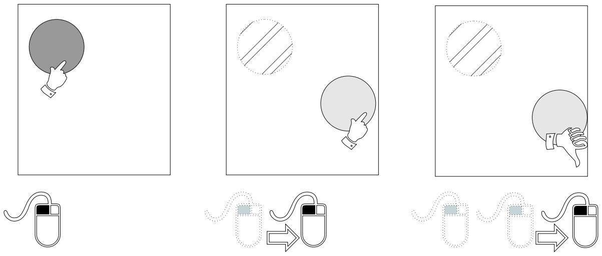


Figure 6. Concept must stay in context.

Figure 6 shows a extreme situation where the concept can not leave its context. The mouse, used as the physical device, is shown. Its movement does not change the distance between conjectural component and logical device when the concept reaches the context boundary. However, the logical device shape changes acknowledging the mouse movement.

A similar behavior is presented when handling the other language elements. In addition, other situations and manipulation of groups are handled as well. A detailed description of this metaphor can be found elsewhere [8].

4.4. The membrane metaphor

In the membrane metaphor, the outline of dominant object is distorted as if it was a membrane. The distortion is applied until a threshold is reached. The strength of the constraint imposed by the dominant is conveyed by the size of the area distorted and again by the size of the field.

Figure 7 represents a situation where an association is defined between two concepts. The exit and enter thresholds are shown as dot-dash ellipses in the origin and destination concepts. Thresholds are visible only when they active. The beginning of the fields coincides with the objects outline.

In a), the concept outline is distorted until the logical device reaches the exit threshold. Thereafter, the conjectural component of the association is moved. Its dotted appearance reflects that it will not be created until a destination is found. The small plane indicates that the manipulation is flying over the diagram space in search for another concept. There is no tangible component, since the association has not been created yet.

When the logical device enters a rejection field of a destination object (in figure 7c), the distortion of its outline starts. Once the enter-threshold is reached the distortion ends. At that point the association may be created and the conjectural component changes its shape (from dotted to a solid arrowhead line).

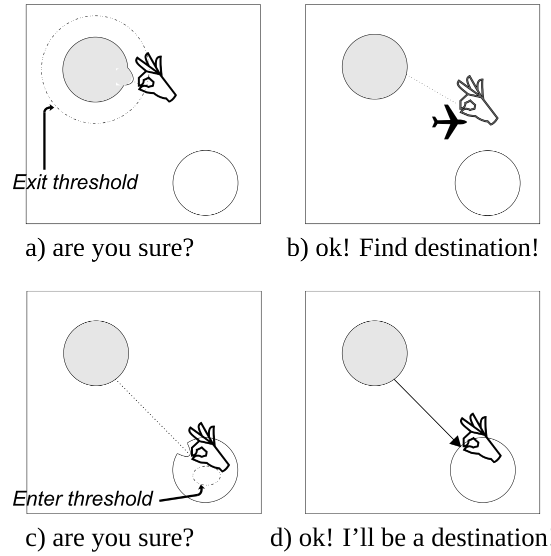


Figure 7. Reluctance to the definition of associations.

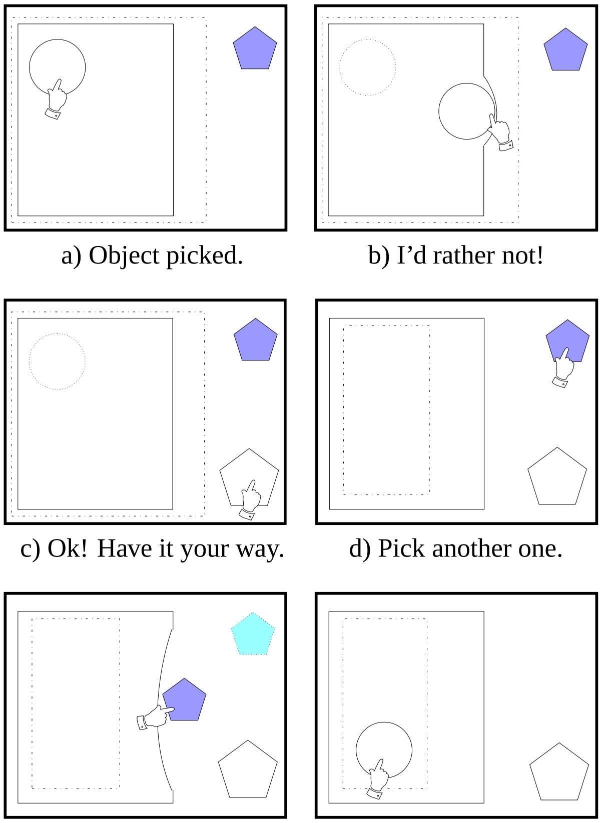
Figure 8 depicts a situation that conveys different influence forces. The elements involved (two concepts and two contexts) are similar to those described on the example of figure 5. The difference is that here the manipulation follows the membrane's metaphor rather than the magnet one. In addition, a manipulation entering a context (parts d) e) and f)) is also shown. The rejection field is defined by the outline of the context and the dot-dash rectangle inside the context.

Besides a distortion of the outline of the context, it is worth noting the effect of the different strengths of the attraction and rejection fields: smaller for attraction than for rejection. That difference is conveyed by the size of the distorted area during manipulation. For the attraction field, the distorted area is smaller than for the rejection field.

Figure 9 shows the classification of a concept on a non-adjacent context. In b) a teleport is used and its boundary deflects proportionally to the target constraint.

5. Related work

Visio [24] and Hardy [2] provide very complete solutions for the manipulation and visualization of diagrams. Standard structuring layouts, emphasizing visual attributes and multiple views over the same diagram are available. Both tools offer mechanisms for the definition of constraints that determine the behavior of nodes and links. Hardy although

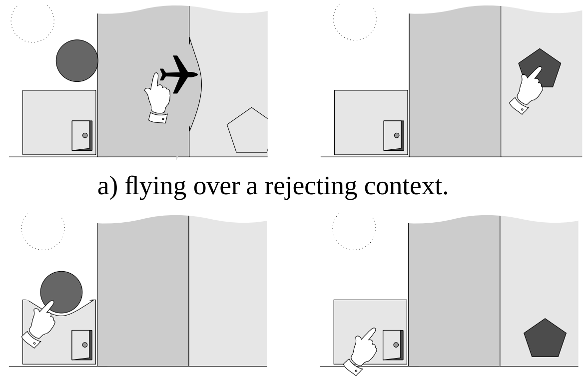


a) Object picked. b) I'd rather not!

c) Ok! Have it your way. d) Pick another one.

e) I don't like the object! f) Ok! I'll have it.

Figure 8. Resistance to the classification of objects within contexts.



a) flying over a rejecting context.

b) Using a teleport.

Figure 9. Flying to non adjacent contexts.

visually less powerful, provides access to rule-based engines. KMap [13] also supports concept maps, access to knowledge base engines and integration with common hypermedia navigation tools. Finally, DecisionExplorer [4] further approaches the application domain of our work. It is designed for the study of cognitive maps of decision-making processes. However, none of the above tools provides support for effective visual feedback. Feedback dialects are static, very simple and sometimes compensated with natural language phrases.

Relating to the proposed metaphors for feedback dialects, some points of contact can be found in current window managers. In particular, `fvwm` already conveys simple forms of resistance. When a manipulated window reaches the boundaries of the physical screen, the mouse cursor stops. If the physical mouse keeps moving, a threshold is attained and a new virtual space is entered.

On another strand of work, Penz and Carriço [17] proposed the inclusion of active and sensitive areas on graphical objects. The model allows objects to "feel" each other and react accordingly. In computer supported cooperative work Benford and Fahlén [5] proposed an object model that includes an aura, a focus and a nimbus area. These areas determine the interaction between objects on a specific medium, managing conversations between several people. Rejection fields of the parents' model offer an approach similar to active areas (on the first model) and nimbus (in the second one). However, the definition of a dual attraction field and the introduction of forces (eventually related to awareness) allow an easier construction feedback dialects, not aimed in the other works.

6. Conclusion and future work

The work presented in this paper emerges from the requirements identified in the development and usage of tools that use cognitive maps [7, 6]. Cognitive maps are expressions of visual languages aiming at the comprehension of reasoning. The syntax and semantic of such languages imposes constraints on their visual elements. When these elements are manipulated, the user expects to have unambiguous semantic feedback of its actions. However, the complexity of constraints can only be conveyed if feedback languages are based in a richer vocabulary.

This paper presents the language elements, actions and feedback required by the specification and manipulation of cognitive maps. It proposes an object dependency model (the parent's model) that deals with variable attraction and rejection fields associated to the constraints that dominant objects impose to their dependents. These constraints and their strengths are related to the uncertainty of the definition and classification of the knowledge represented in the cognitive maps. The proposed system also supports two feedback dialects. These dialects, using the generic properties of object dependency model, convey resistance to the direct manipulation of cognitive maps' elements.

Our current work is focusses on the empirical evaluation of the visual language and the feedback dialects. This is being done within an ongoing project, named Cognitive Mapping of the Negotiation Processes, that aims at studying the cognitive maps of negotiators before, during and after negotiations. The project, from the technological point of view, adapts an existing cognitive mapping tool [15] and extends it with specific fuzzy logic techniques plus the elements and feedback techniques described in this paper.

Future plans include integration with a group decision support system [1]. The highly interactive nature of group support systems will provide a valuable contribution to the evaluation.

Acknowledgements

This work was partially developed within the project Cognitive Mapping of the Negotiation Processes, PRAXIS/PCSH/P/PSI/77/96, funded by the PRAXIS XXI program.

References

[1] P. Antunes and N. Guimarães. Structuring elements for group interaction. In *Second Conference on Concurrent Engineering, Research and Applications (CE95)*, Washington, DC, Aug. 1995. Concurrent Technologies Corporation.

[2] Artificial Intelligence Applications Institute. Hardy homepage, 1997. `http://www.aiai.ed.ac.uk/~hardy`.

[3] R. Axelrod. *Structure of Decision: The Cognitive Maps of Political Elites*. Princeton University Press, Princeton, New Jersey, 1976.

[4] Banxia Software. Decision Explorer homepage, 1997. `http://www.scotnet.co.uk/banxia/demaim.html`.

[5] S. Benford. A spatial model of interaction in large virtual environments. In *Proceedings of the Third European Conference on Computer-Supported Cooperative Work – ECSCW '93*, Milan, Sept. 1993.

[6] L. Carriço, P. Antunes, and N. Guimarães. Managing perspectives of changing organizational systems. Document available at `http://di.fc.ul.pt/~lmc`, 1997.

[7] L. Carriço and N. Guimarães. Facilitating analysis and diagnosis in organisations. In *Proceedings of the CAiSE '97 Conference*, Barcelona, Spain, June 1997.

[8] L. Carriço and N. Guimarães. Manipulating concept maps with constrained regions. In *Proceedings of the AVI'98 Conference*, L'Aquila, Italy, May 1998.

[9] L. Carriço and N. Guimarães. Integrated multiviews. *Journal of Visual Languages and Computing, Special Issue on Visual Navigation: Methods and Tools*, 9(3):287–297, June 1998.

[10] J. Conklin. Gibis: a hypertext tool for exploratory policy discussion. In *Proceedings of the Conference on Computer-Supported Cooperative Work (CSCW '88)*, pages 140–152, Portland, Oregon, 1988. ACM Press.

[11] C. Eden. Strategy development and implementation: Cognitive mapping for group support. In J. Hendry and G. J. with J. Newton, editors, *Strategic Thinking: Leadership and the Management of Change*. Addison-Wesley, 1993.

[12] G. Fisher, R. McCall, and A. Morch. JANUS: Integrating hypertext with a knowledge-based design environment. In *Hypertext 89*, Nov. 1989.

[13] B. Gaines and M. Shaw. Concept maps as hypermedia components. *Knowledge Science Institute Home Page*, 1995. `http://ksi.cpsc.ucalgary.ca`.

[14] A. Huff. Mapping strategic thought. In A. Huff, editor, *Mapping Strategic Thought*. John Wiley & Sons, Ltd., 1990.

[15] M. Laukkanen. *Comparative Cause Mapping of Management Cognitions*. Helsinki School of Economics and Business Administration, 1992.

[16] C. Neuwirth and D. Kaufer. The role of external representations in the writing process: Implications for the design of hypertext-based writing tools. In *Hypertext 89*, 1989.

[17] F. Penz and L. Carriço. Objects feeling objects in a multiview object space. In *Vienna Conference on HUMAN CONPUTER INTERACTION 93, Fin de Siècle*, Viena, Austria, Sept. 1993.

[18] J. Porras. *Stream Analysis*. Organization Development. Addison-Wesley, 1987.

[19] R. Reger. Managerial thought structures and competitive positioning. In A. Huff, editor, *Mapping Strategic Thought*. John Wiley & Sons, Ltd., 1990.

[20] L. Shastri. *Semantic Networks: An Evidential Formalization and its Connectionist Realization*. Morgan Kaufmann Publishers, Inc, 1988.

[21] S. Shum and N. Hammond. Argumentation-based design rationale: From conceptual roots to current use. Technical Report EPC-1993-106, Rank Xerox Research Centre, 1993.

[22] J. Sowa, editor. *Principles of Semantic Networks: Explorations in the Representation of Knowledge*. Morgan-Kaufman, San Mateo, California, 1991.

[23] N. Streitz, J. Hannemenn, and M. Thuring. From ideas and arguments to hyperdocuments: traveling through activity spaces. In *Hypertext 89*, 1989.

[24] Visio. Visio homepage, 1997. `http://www.visio.com`.