

2016 ICCRTS Information Central

London, UK

September 6-8, 2016

Main topic of paper: Methodological Development, Experimentation, Analysis,
Assessment and Metrics

Knowledge Representation in Experiential Learning

Sarah Markham (PhD Student)

Department of Computer Science and Information Systems

Birkbeck College London

smarkham@dcs.bbk.ac.uk

Abstract

In organisations peopled by expert individuals with specific experiential knowledge bases, the knowledge they have which is of relevance to the organisation may be lost when they leave it. Some of this knowledge may be tacit and not explicitly requested, expressed or recognised. This paper presents an exploration of potential mechanisms of expert experiential knowledge identification, capture and representation with a view to automated reasoning, sharing and handover within organisations.

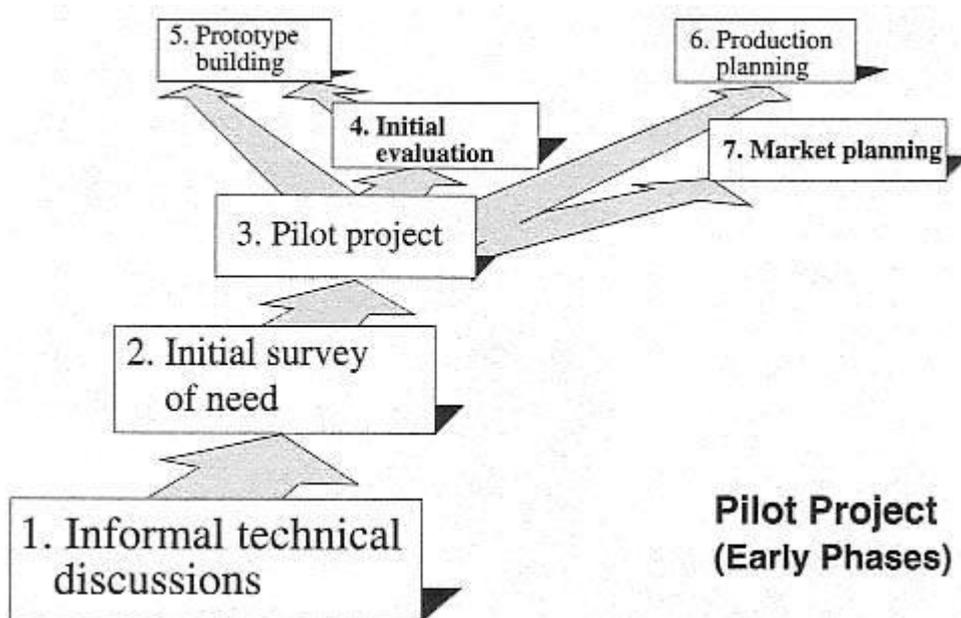
Keywords

Experiential, Learning, PERT, Abstract, Argumentation, Framework

1. Graphical Representation for Systems Support

Studies in cognition [1, 2] have investigated the role of external visual representations in different domains in supporting reasoning, problem solving, and communication. In [3] Dogan and Nersessian remark that in many studies of well-defined problems, diagrammatic representations illustrate either causal or temporal relationships between parts of entities and phenomena that the diagram represents.

Illustrating a dynamic process with an arrow-containing diagram is a widespread convention in daily communications. (See the diagram below illustrating the temporal ordering of the early phases of a Pilot Project.)



In [4] Engelhardt suggests regarding the building blocks of all graphics as falling into three main categories: a) the graphic objects that are shown (e.g., a dot, a pictogram, an arrow), b) the meaningful graphic spaces into which these objects are arranged (e.g., a geographic coordinate system, a timeline), and c) the graphic properties of these objects (e.g., their colours, their sizes).

Graphic objects come in different syntactic categories, such as nodes, labels, frames, links, etc. Such syntactic categories of graphic objects can explain the

permissible spatial relationships between objects in a graphic representation. In addition, syntactic categories provide a criterion for distinguishing meaningful basic constituents of graphics. Various syntactic principles can be identified in graphics of different types, and the nature of visual representation allows for visual nesting and recursion.

Graphics can be regarded as expressions in visual languages. Engelhardt proposes that specifying such a visual language means a) specifying the syntactic categories of its graphic objects, plus b) specifying the graphic space in which these graphic objects are positioned, plus c) specifying the visual coding rules that determine the graphic properties of these graphic objects. The syntactic structure of a graphic representation is determined by the rules of attachment for each of the involved syntactic categories and by the structure of the meaningful graphic space that is involved. Engelhardt proposes a limited set of possible 'building blocks' for constructing graphic spaces, and a limited set of possible syntactic functions of graphic objects. Based on these ingredients, and the rules for their combination, the syntactic structure of any visual representation can be drawn as a hierarchically nested tree.

In other words a set of graphic objects can be combined into a meaningful arrangement, together forming a single graphic object at a higher level. As Winn writes in [5]: "One property of the symbol system of maps and diagrams is that their components can form clusters, which in turn can form other clusters in a hierarchical fashion.

In [6] Barker-Plummer gives an overview of this formal perspective on diagrams, and to introduce and explain the techniques used by logicians to analyse reasoning with diagrammatic representations. In his view the three main questions are expressive completeness; the degree to which diagrams can be used to represent information about a given domain, soundness of inference; how we can guarantee the validity of conclusions reached by reasoning with diagrams, and completeness of inference; the range of conclusions that can be reached by using those techniques.

2. Knowledge Representation for Planning with Capture of Experiential Learning

Research in knowledge representation for intelligent support has focused on syntactical reasoning over semantic reasoning [7]. In domains which tend

towards abstract and/or philosophical and/or theoretical scientific contexts and representation, syntactical reasoning is adaptively appropriate. However in the more pragmatic and immediate domains of say planning military operations, with the aim of summarising and comparing plans in order to capture and identify experiential learning, semantic reasoning modalities would appear more useful in helping users to navigate more effectively through large solution spaces to identify plans and tactics that are well-suited to their needs.

Domain metatheory provides the potential to abstract from the details of plan structures to concise summarizations of key decisions within plans, and to important implemental changes within plans. In [8] and [9] Myers and Lee describe an approach that employs a suite of techniques to identify patterns or exceptions relative to meta-theoretic structures such as role, task or feature abstraction. Myers approach is limited in that the visual representation is restricted to tabular forms which give little indication of sequential dependency between tasks and a paucity of syntactic analysis/reasoning. Myers approach also neglects the capture/representation of plan modification and experiential learning.

From the AI planning point of view, depending on how it is approached, visualisation can play two main crucial roles in planning: (1) to permit collaboration among participant agents in the case of collaborative planning systems; (2) to allow proper interfacing between the software and human planners. What has hitherto been neglected is the potential for visualisation to potentiate and facilitate experiential learning via capture of plan modification in implementation.

To address this problem, I propose a general framework for visualisation in planning systems that will give support for a more appropriate visualisation mechanism based to some extent on that described by Correia Queiroz in [10], but with specific adaptation and extension to experiential learning and abstract argumentation as an aid to utilising experiential learning in the formulation of further plans.

This framework is divided into four main parts:

- 1) a knowledge representation aspect
- 2) an identification and recording mechanism for experiential learning

- 3) plan retrieval and comparison facility
- 4) graphical support for reasoning in plan formation (using past plans and experiential learning) in the form of an argumentation framework.

I envision creation of a knowledge base which lends itself to knowledge representation for intelligent support in environments oriented towards experiential learning with continuous modification.

The envisioned knowledge base will consist of the following elements:

Knowledge acquisition – Agents record plans of operations before implementation in the ‘planning stage’, during and after implementation, together with reasons for any modifications to the initial plan that occurred during plan actualisation. This will require some form of agent/knowledge base interface to facilitate knowledge acquisition.

Knowledge representation – Complex projects require a series of activities, some of which must be performed sequentially and others that can be performed in parallel with other activities. This collection of sequential and parallel tasks can be modelled as a network. The Program Evaluation and Review Technique (PERT) is a network model that allows for randomness in activity completion times. PERT was developed in the late 1950's for the U.S. Navy's Polaris project having thousands of contractors. It has the potential to reduce both the time and cost required to complete a project. Using PERT diagrams to provide a visual representation of operational plans capturing both pre-operation aspirations and intentions and post-operation actuality and outcomes will allow experiential learning to be recorded and identified. The PERT chart may have multiple pages with many sub-tasks. To facilitate ease of representation, retrieval and learning, sub-tasks (on separate pages) can be annotated to any desired degree with a comparison between intended and actualised plans in terms of agency, resources, modality and time.

Plan modifications tagged by:
1) Name of agent
2) Date
3) Reason : rethink/new intelligence/incident during plan implementation...
Recorded in tabular and diagrammatic form.

A graphical reasoning representation mechanism will give support to reasoning about the visualisation problem based on the knowledge bases available for a realistic collaborative planning environment, including agent preferences, device features, planning information, visualisation modalities, etc.

Knowledge inferencing – The facility to interrogate the Knowledge Base using

- 1) Logical Reasoning – predicate (declarative) queries
- 2) Deductive Reasoning – IF {conditions} then DO {actions} captured in instances of experiential learning. This would require rules to be input, added and modified.
- 3) Analogical Reasoning – maximal isomorphic subgraph identification to capture similarities in syntactic structure – matching sub-networks of tasks in individual PERT Diagrams (relevant in military planning contexts).

In Appendix B we present candidate algorithms for the identification of edge-matched maximal isomorphic subgraphs of distinct graphs with extension to analogical reasoning.

4) Knowledge transfer to the user – form or output dependent on reasoning modality and input (question), for example:

- 1) Predicate validation
- 2) Outcomes (Actions & Effects)
- 3) Maximal Isomorphic Graph Theoretic Relation Representation –to capture similarities in syntactic structure.

This plan summarization, analysis and comparison method is domain-independent, making it applicable to a broad range of problems. In particular, it avoids domain-specific algorithms or bodies of knowledge that would limit the applicability of the method.

The value of the domain metatheory lies with its provision of a semantically-grounded abstraction facility with semantics to be pre-specified – i.e. roles/tasks/role and task attributes/categories of attributes with customisable fields of all types). For all (atomic) tasks, there will exist (modifiable) associated roles and the facility to assign modifiable attributes to the atomic tasks and

associated roles. Task attributes capture important semantic attributes of a task. Task attributes are modelled in terms of an attribute category and value. A role describes a capacity in which an individual is used within a task; it maps to a task variable. Roles also provide the means to reference a collection of semantically linked variables that span different contexts and tasks. As part of the abstraction process, the metatheory will provide semantic linkage among different elements within a planning domain. Semantic linkage enables descriptions of plan properties, both on the part of a user seeking to direct a planning system, and a system seeking to summarize plans or planning decisions for a user.

Experiential learning advice once captured could be recorded in the form:

<Use/Don't Use> <resource> in <role>for <context-task>

In general, experiential learning advice consists of:

- 1) one or more specified roles with associated attributes/resources
- 2) a contextual task
- 3) a polarity indicating whether the advice is prescribing or prohibiting the associated attributes/use of resources.

The value of task and role attributes for plan summarization and comparison is that they provide the means to identify, abstract, and contrast important evaluative properties or measures of different strategies, such as speed or risk. A measure corresponds to an ordering (possibly partial) of features within the category with respect to some designated criteria. The range of a measure is the set of (partially) ordered values employed by the measure.

For measures defined over attribute categories, the domain is the set of attributes that comprise the feature category. For measures defined over instances of plan implementation, the domain is the set of measure values that can be assigned to instances.

One natural way to support the use of automated planning and search technology is to allow users to direct the operations of the underlying planning search by specifying desired plan attributes and measure values. For example, consider the attribute category terrain with attributes {hidden mines, water-logged}. For the measure RISK, the attribute 'water-logged' would rank lower than 'hidden mines'. An agent planner could express preferences for a particular operation in terms of risk and ease of crossing of terrain with an

automated planner constructing a solution that seeks to maximize satisfaction of those preferences according to pre-specified attributes.

3. Graphical Representation for Abstract Argumentation Support in Plan Formation incorporating Experiential Learning

Understanding argumentation and its role in human reasoning has been a continuous subject of investigation for scholars from the ancient Greek philosophers to current researchers in philosophy, logic and artificial intelligence. In recent years, argumentation models have been used in different areas such as knowledge representation, explanation, proof elaboration, common sense reasoning, logic programming, legal reasoning, decision making, and negotiation. However, these models address quite specific needs and there is need for a conceptual framework that would organize and compare existing argumentation based models and methods. Such a framework would be very useful especially for researchers and practitioners who want to select appropriate argumentation models or techniques to be incorporated in new software systems with argumentation capabilities.

In [11] such a conceptual framework is proposed, based on taxonomy of the most important argumentation models, approaches and systems found in the literature. This framework highlights the similarities and differences between these argumentation models.

Arguments can be considered as tentative proofs for propositions. In formal argumentation, knowledge is expressed in a logical language, with the axioms of the language corresponding to premises according to the underlying domain. Theorems in the language correspond to claims in the domain which can be derived from the premises by successive applications of some inference rules.

Generally, the premises are inconsistent in the sense that contrary propositions may be derived from them. In this formulation, arguments for propositions, or claims, are the same as proofs in a deductive logic, except that the premises on which these proofs rest are not all known to be true. The understanding of an argument as a tentative proof and a chain of rules attends to its internal structure. Several models addressing the internal structure of arguments have been developed. These models stress the link between the

different components of an argument and how a conclusion is related to a set of premises. They mainly consider the relationships that can exist between the different components of an argument in a monological structure. For this reason, we call the models belonging to this category: monological models.

A second strand of research in artificial intelligence has emphasized the relationships existing between arguments, sometimes considered as abstract entities and ignoring their internal structures. Because they highlight the structure of arguments as presented in a dialogical framework, the models belonging to this category are called dialogical models.

Generally, monological models and dialogical models consider respectively the internal (micro) and external (macro) structure of arguments. While dialogical models and rhetorical models of argumentation highlight the process of argumentation in a dialogue structure, monological models emphasize the structure of the argument itself. What is important in these models is not the relationship that can exist between arguments, but the relationships between the different components of a given argument.

To model the notions of arguments Reed and Walton [12] proposed the notion of argumentation schemes. Argumentation schemes are the forms of arguments describing the structures of inference. This notion enabled the authors to identify and evaluate common types of argumentation in everyday discourse. Such schemes can be used to represent knowledge needed for arguing and explaining.

Argument schemes are not classified according to their logical form but according to their content. Many argument schemes in fact express epistemological principles (such as the scheme from the expert opinion) or principles of practical reasoning (such as the scheme from consequences). Accordingly, different domains may have different sets of such principles.

Each argument scheme comes with a customized set of critical questions that have to be answered when assessing whether their application in a specific case is warranted. Clearly, the possibility to ask such critical questions makes argument schemes defeasible, since negative answers to such critical questions are in fact counterarguments, such as “Expert E is not sincere since he is a relative of the suspect and relatives of suspects tend to protect the suspect”.

Reed et al. [13] developed a system, called Araucaria System (see Section 5) in order to construct an online repository of arguments drawn from newspaper editorials, parliamentary reports and judicial summaries. It is such a framework that we envisage for supporting plan formation using experiential learning via an abstract argumentative framework.

Monological models of argumentation focus on structural relationships between arguments. It is this approach that we will take. In [14] a theoretical framework for a general diagrammatic literacy is presented, based on conceptualizing diagrams in terms of function rather than form. Approaching diagrams functionally generates a framework for thinking critically about diagrams (in general) that is simple, robust and exhaustive. In addition to this functional approach, the role of context and language to the internal definition of any given diagram is emphasized.

Previous research has highlighted the advantages of graphical argument representations. A number of tutoring systems have been built that support students in rendering arguments graphically, as they learn argumentation skills [15]. Researchers aiming to develop systems that engage students in argument or improve their argumentation skills have been drawn to graphical representations for a number of reasons. From a cognitive perspective, graphical representations can reduce the students' cognitive load and reify important relationships. Thus, it is hypothesized, they facilitate reasoning about texts and the acquisition of interpretive skills [16, 17].

While the use of two simultaneous representations can increase cognitive load, the complementary strengths of a textual and graphical argument form can better guide students in their analysis. Second, intelligent tutoring systems can provide feedback on graphical argument representations while finessing the fact that natural language processing remains difficult. A student-made graph provides the system with information about their thinking that, even if it does not rise to the level of complete understanding, can be leveraged to provide intelligent help [18].

[19] addresses some open research questions, including how to model the process of hypothetical reasoning in order to explain its role in legal argument, how to implement the process computationally for purposes of teaching students, and how to evaluate such a model. A process model of arguing with hypotheticals and demonstrating how it accounts in a natural way for common features of SCOTUS legal arguments. It is explained how, and the extent to

which, the process model has been implemented computationally in the LARGO (Legal ARgument Graph Observer) intelligent tutoring system. The program supports students in diagramming SCOTUS oral argument examples in accord with the process model; its feedback on students' diagrammatic reconstructions of the examples enforces the model's expectations.

4. Formalisation of the Use of PERT Diagrams in Capturing Experiential Learning

The objective is to exploit the underlying digraphic structure of PERT Diagrams to represent plans. This will involve tasks, implemented in sequence and in parallel, being represented by labelled directed edges and corresponding series of events represented as labelled nodes. Modifications to a plan during periods of execution will lead to the creation of a series of corresponding graphs or (according to user preference) extensions to original plan.

These changes will also be represented textually as extensions to corresponding tasks (edge labels) and events (node labels) detailing the nature of and reasons for the individual modifications. These textual additions will be made to both the members of each corresponding PERT Diagram pair (in the case of generation of a temporal series of modified PERT Diagrams) or each corresponding node and edge pair (in the case of an extended original graph).

Vertices represent events:

Vertex Set = { Event r : Event 0 is the Start, Event n is the Finish, $n > 0$,
 $r = 1, \dots, n$ }

Directed Edges represent tasks:

Edge Set = {Task t : $t = 1, \dots, m$ }

The directional arrows (edges) represent tasks to be completed sequentially over time. Diverging edges indicate possibly concurrent tasks (to be implemented in parallel).

Rectangles represent nodes allowing the following data to be recorded:

- 1) Desired outcome or event
- 2) Modified outcome or event (plus reasons for modification) or Actual outcome or event (plus reasons for difference from final plan)

NB

The rectangle representing the START node allows the following data to be recorded:

Goal of Plan

Scheduled start

Modified scheduled start (plus reason for modification)

or

Actual start (plus reason for difference from plan)

Rectangle representing the FINISH node allows the following data to be recorded:

Actual outcome(s)

Scheduled finish

Modified scheduled finish (plus reason for modification)

or

Actual Finish (plus reason for difference)

Directed edges are labelled with rectangles in which the corresponding task details are recorded:

Scheduled Start

Actual Start

Modified Scheduled Finish &

Actual Finish

or

Scheduled Finish &

Actual Finish

Reasons for difference

Scheduled Roles

Modified Roles

or

Actual Roles

Reasons for difference

Role Attributes (Scheduled)

Role Attributes (Modified) or Role Attributes (Actual)

Reasons for difference

Sub-Tasks (Scheduled)

Sub-Tasks (Modified) or Sub-Tasks (Actual)

Reasons for difference

Sub-Task Attributes (Scheduled)

Sub-Task Attributes (Modified) or Sub-Task Attributes (Actual)

Reasons for difference

Resources (Scheduled)

Resources (Modified) or Resources (Actual)

Reasons for difference

Environment (Forecast)

Environment (Modified) or Environment (Actual)

Reasons for Difference

To indicate when the actual operation has diverged from the current version of the plan (to the extent that different tasks are implemented and possibly different outcomes or events sought and/or obtained):

1) New event rectangles represented by differently coloured rectangles/nodes and corresponding new coloured task rectangle/edge labels will be deployed to capture the various stages in the evolution of the plan as it is implemented and changes are necessitated.

2) Alternatively new PERT Diagrams will be created showing the actual as opposed to planned events and tasks. This will lead to the creation of generations of plans over time with different colours used to highlight the changes that occur between each generation. There will be facility to view (compare and contrast) successive PERT Diagrams via one "evolvable" window (e.g. with a time slider).

In Appendix A we present an example of a template for a PERT Diagram Representation of a plan together with modifications recorded to capture experiential learning

A database of past PERT Diagram plan representations will be used as a planning aide:

The database will be searched according to ontological tags describing the characteristic events, roles, resources and other features or attributes of each individual historical plan. The retrieved plans will be ordered according to a relevant similarity metric in ascending order with the most similar to the desired plans tentative (military ontological) characteristics given priority.

When deciding on which metric is most relevant and representative as a measure of the similarity or difference between two plans, we examine how we wish to compare (intended or actualised) events, roles, resources and tasks both at base and attributional level. The following measures tentatively capture this, the second (semantic) metric doing so with use of specific weights to represent the relative contribution of different tasks, roles, resources used and environmental attributes, to divergence from similarity.

The value of task and role attributes for plan summarization and comparison is that they provide the means to identify, abstract, and contrast important

evaluative properties or measures of different strategies, such as speed or risk. For measures defined over attribute categories, the domain is the set of attributes that comprise the feature category. For measures defined over instances of plan implementation, the domain is the set of measure values that can be assigned to instances.

We suggest the following metrics as a tentative measures of plan structural and plan semantical similarity. In doing so, in the case of structural similarity we consider the underlying graph-theoretic nature of plans, i.e. the plans as being formed of node and edge sets.

Let $G(V,E)$ and $G'(V',E')$ be two graphs with adjacency matrices A and A' respectively corresponding to plans P and P' . Given the structure of the graph (and therefore the corresponding plan) is completely defined by the adjacency matrix, we define the following structural metric on G and G' where n (n') and m (m') are the number of nodes and edges of G (G'):

Let A^* be the extension of adjacency matrix A formed by adding rows and columns of zeroes to A until we have a matrix with n^* rows and m^* columns, where $n^* = \max(n, n')$ and $m^* = \max(m, m')$

Similarly let A'^* be the extension of adjacency matrix A' formed by adding rows and columns of zeroes to A' until we have a matrix with n^* rows and m^* columns

$$\text{syn}(P, P') := d(G^*, G'^*) = \text{Sum } |A^*_{ij} - A'^*_{ij}|$$

where the sum is over all the indices $i: 1 \leq i \leq n^*, 1 \leq j \leq m^*$

We define a semantic metric on P and P' by considering the similarities and differences in the sets of tasks (T, T'), roles (S, S'), resources (R, R') and environmental attributes (E, E') in the individual plans:

$$\text{sem}(P, P') :=$$

$$w_1 |T \setminus T'| + |T' \setminus T| - |T \cap T'| + w_2 |S \setminus S'| + |S' \setminus S| - |S \cap S'| + w_3 |R \setminus R'| + |R' \setminus R| - |R \cap R'| + w_4 |E \setminus E'| + |E' \setminus E| - |E \cap E'|$$

$T \setminus T'$ is the set of tasks in T but not T' , etc

$T \cap T'$ is the set of tasks in T and T' .

With similar definitions for S, R, E , etc.

There will also be a graphical argument facility to help determine how best to use the stored plans and associated experiential learning data to inform future plan formation and plan critique, with a view to determining the most adaptive and robust plans for future operations. This will use a Toulmin model [20] based diagram to graphically represent and formulate tests of the applicability of instances of experiential learning to the formation of individual plans and/or test the integrity of given plans in the light of (new) pieces of experiential learning. A more detailed description of the argument theoretic planning aide is given in the next section.

5. Araucaria

In [21] the Araucaria program for graphical representation of abstract argumentation frameworks, is composed of three main sections:

- 1) A main window which allows argument diagrams to be constructed from pre-existing text files.
- 2) An editor for schemes and scheme sets.
- 3) An interface to an AraucariaDB online repository of marked up arguments.

When Araucaria loads, the program displays its main window which can be used to load text files and create argument diagrams from the text. When a text file is loaded, the text appears in the left-hand panel. A portion of this text may be selected with the mouse. If the mouse is then clicked in the large panel on the right, a node corresponding to that portion of the text is created and drawn at the bottom of the panel. When two or more nodes have been defined in this way, they can be connected in pairs by selecting one node with the mouse and dragging the mouse to the other node. The first node selected is the premise of an argument, and the second node is the conclusion.

In addition to the features described above for inserting components into a diagram, Araucaria allows components to be deleted from a diagram, and also contains full 'undo' and 'redo' capability. As preference seems divided in both the research and pedagogic communities, the entire diagram can also be inverted. Finally, all analyses can be saved and loaded, and diagrams can be exported as JPEG images.

Each node and support arrow in a diagram can also have an associated evaluation, which can be used to represent the confidence placed in a premise or support. To attach an evaluation to one or more parts of the diagram, the nodes and/or support arrows are selected and the evaluation editor is used to define the associated evaluation. Evaluations are displayed as labels next to the node or arrow on the main diagram.

Schemes and Scheme Sets

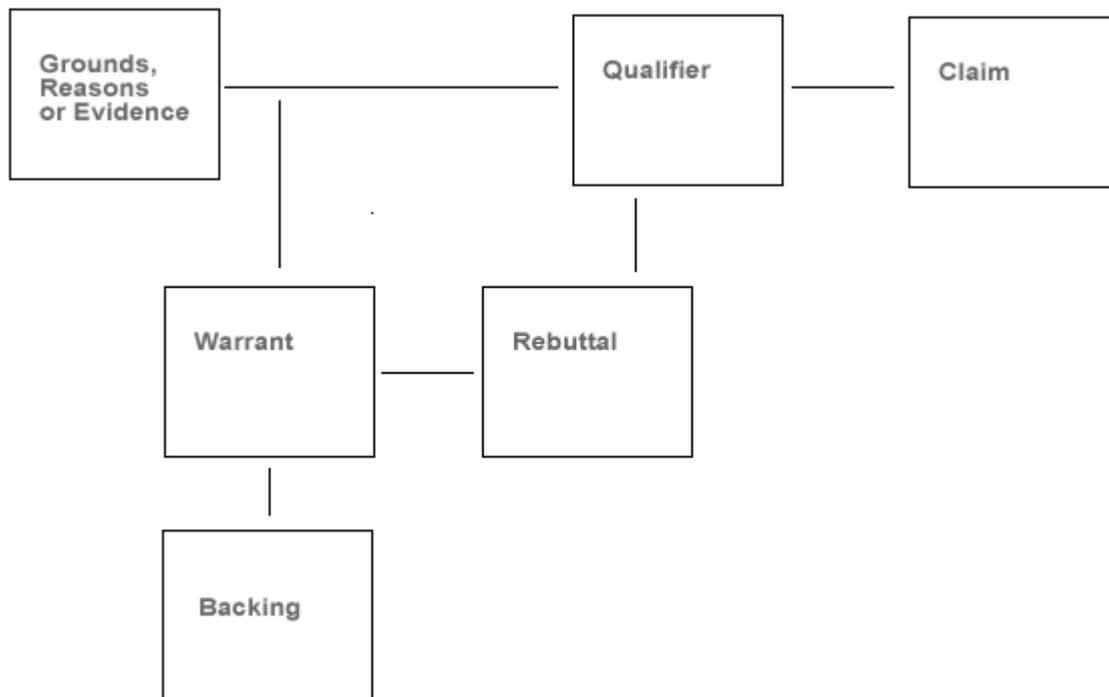
Araucaria allows the user to define argumentation schemes and to group them together into scheme sets. The scheme editor allows a scheme to be defined by specifying its name, conclusion, premises and critical questions. A scheme set containing a number of schemes can then be saved in a scheme set file.

AraucariaDB Online Repository

AraucariaDB is an online database of marked up arguments maintained at the University of Dundee. Araucaria provides an interface via the internet to AraucariaDB, which allows users to search the database for arguments using several search criteria, and also to add their own marked up arguments to the database. A web-based interface that obviates the need for Araucaria for read access is also available.

We propose an abstract argumentation planning aide similar to Araucaria, but tailored to support users in the deployment of past plans and associated experiential learning in future plan formulation. When formulating a plan users will be able to search for relevant plans according to ontologically tagged plan objectives, events, tasks and other associated features or attributes. When deciding whether or not to utilise instance(s) of experiential learning, the user will have the option of using an abstract argumentation planning aide based on the Toulmin model (see below) to graphically facilitate their decision making process.

The Toulmin Model



Claim: the position or claim being argued for; the conclusion of the argument.

Grounds: reasons or supporting evidence that bolster the claim.

Warrant: the principle, provision or chain of reasoning that connects the grounds/reason to the claim.

Backing: support, justification, reasons to back up the warrant.

Rebuttal/Reservation: exceptions to the claim; description and rebuttal of counter-examples and counter-arguments.

Qualification: specification of limits to claim, warrant and backing. The degree of conditionality asserted.

The claim will represent the desired objective (mission, operation event) together with the proposed means of actualisation (tasks, roles, resources, etc.)

Instances of experiential learning can be represented as grounds, backing, rebuttal, qualifier as appropriate together with data relating to other (contextual) features of the new plan.

Conclusion

We have proposed a PERT diagram based form of knowledge representation for intelligent planning support in domains oriented to experiential learning (capture). In addition we have outlined a graphical abstract argumentation framework to aid the reasoned utilisation of experiential learning in the formulation of future plans

References

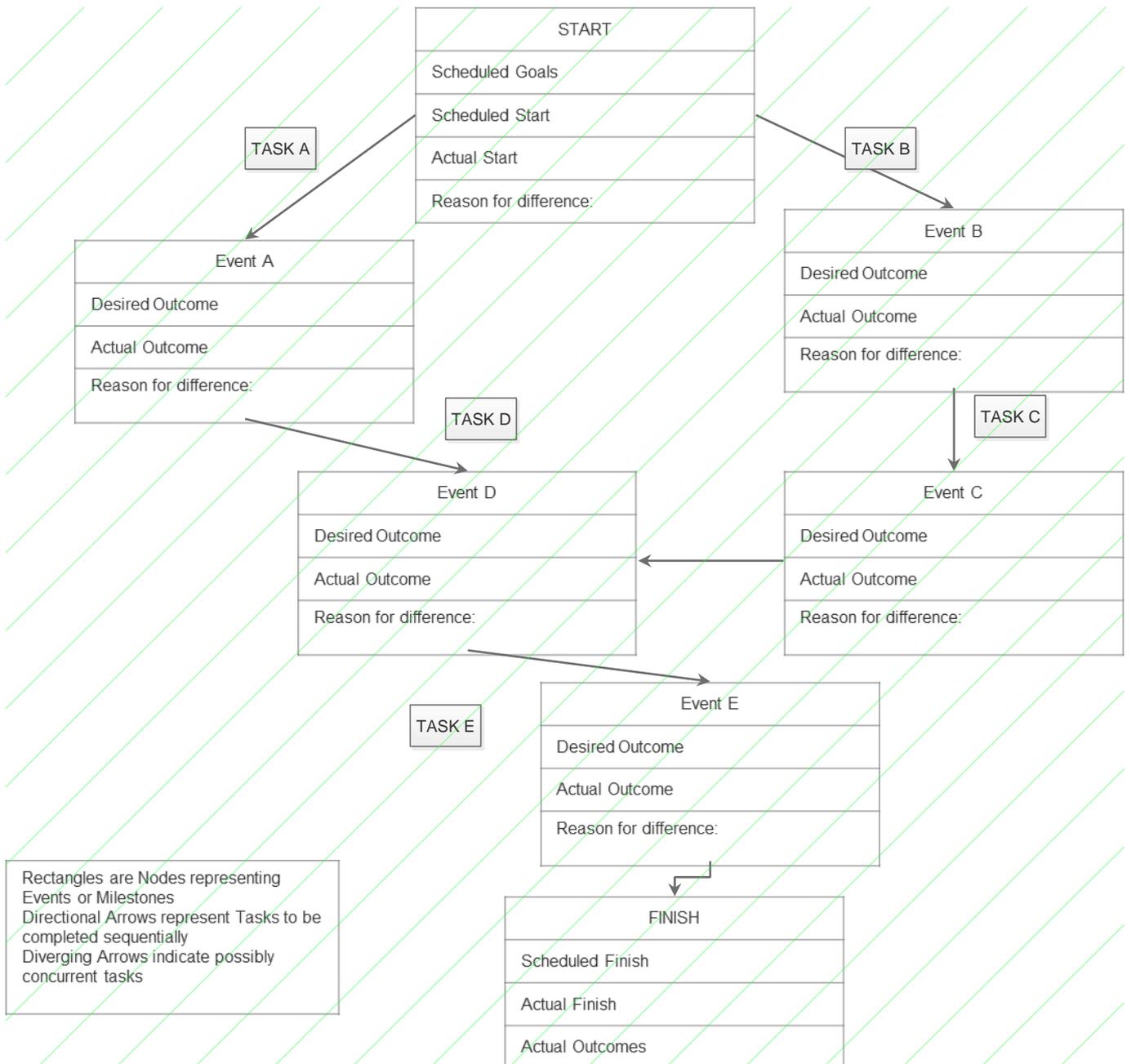
- [1] Zhang, J. (1994). "Representations in distributed cognitive tasks." *Cognitive Science Journal* 18(1): 87-122.
- [2] Larkin, J. H. and H. A. Simon (1987). "Why a diagram is (sometimes) worth ten thousand words." *Cognitive Science* 11: 65-99.
- [3] Conceptual Diagrams: Representing Ideas in Design, Fehmi Dogan¹ and Nancy J. Nersessian², ¹College of Architecture, Georgia Institute of Technology, fehmi.dogan@arch.gatech.edu, ²Program in Cognitive Science, Georgia Institute of Technology
- [4] Yuri Engelhardt Syntactic Structures in Graphics
- [5] Winn, W.: Learning from Maps and Diagrams. *Educational Psychology Review*, Vol. 3 (1991), No. 3, 211- 247.
- [6] Diagrams: A Perspective from Logic, Dave Barker-Plummer, CSLI, Stanford University, dbp@stanford.edu
- [7] 'Young, R. M. 1999. Cooperative plan identification: Constructing concise and effective plan descriptions. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*. AAAI Press.
- [8] Karen L. Myers. 2002. Metatheoretic Plan Summarization and Comparison. AI Center, SRI International 333 Ravenswood Ave. Menlo Park, California 94025 myers@ai.sri.com. American Association for Artificial Intelligence (www.aaai.org).
- [9] Myers, K. L., and Lee, T. J. 1999. Generating qualitatively different plans through metatheoretic biases. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*. AAAI Press.

- [10] Natasha Correia Queiroz Lino. 2007 Doctor of Philosophy. Artificial Intelligence Applications Institute. School of Informatics. University of Edinburgh
- [11] Artif Intell Rev (2010) 33:211–259, DOI 10.1007/s10462-010-9154-1, A taxonomy of argumentation models used for knowledge representation, Jamal Bentahar · Bernard Moulin · Micheline Bélanger, Published online: 19 January 2010 © Springer Science+Business Media B.V. 2010
- [12] Reed C, Walton D (2003) Argumentation schemes in argument-as-process and argument-as-product. In: Proceedings of the conference celebrating informal Logic @25
- [13] Reed C, Rowe G (2001) Araucaria: software for puzzles in argument diagramming and XML. Technical Report, Department of Applied Computing, University of Dundee, Scotland
- [14] Gordon TF (1994) Computational dialectics. In: 1st Workshop on computational dialectics. 12th National conference on artificial intelligence—AAAI'94
- [15] Loui RP, Norman J (1995) Rationales and argument moves. Artif Intell Law 3(3):159–189
- [16] Towards a General Diagrammatic Literacy: An Approach to Thinking Critically about Diagrams, Brad Jackel, Australian Council for Educational Research (ACER), Camberwell, Australia, 3124 Brad.Jackel@acer.edu.au
- [17] Ainsworth, S. (1999). The functions of multiple representations. Computers and Education 33, 131-152.
- [18] Larkin, J. & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. Cognitive Science, 11, 65-99.
- [19] Evaluating Legal Argument Instruction with Graphical Representations using LARGO, Niels Pinkwart¹, Vincent Aleven², Kevin Ashley³, and Collin Lynch⁴, ¹Clausthal University of Technology, Computer Science Institute, Germany, ²Carnegie Mellon University, HCI Institute, Pittsburgh PA, USA, ³University of Pittsburgh, LRDC and School of Law, Pittsburgh PA, USA, ⁴University of Pittsburgh, Intelligent Systems Program, Pittsburgh PA, USA
- [20] Toulmin Model of Argument
<https://web.cn.edu/kwheeler/documents/Toulmin.pdf>

[21] Araucaria: Software for Argument Analysis, Diagramming and Representation, Chris Reed & Glenn Rowe, Supporting learning through intelligent and socially informed technology, vol. 125, pp. 435–442. IOS Press, Amsterdam (2005)

Appendix A

PERT Diagram Representation of Plan and Modifications to Capture Experiential Learning



Task on Arrows PERT Diagram (lends itself to analogical reasoning)

Task Detail Template:

TASK A		
Scheduled Start	Actual Start	Reason For Difference
Scheduled Finish	Actual Finish	Reason For Difference
Scheduled Roles	Actual Roles	Reason For Difference
Scheduled Role Attributes	Actual Role Attributes	Reason For Difference
Scheduled Sub-Tasks	Actual Sub-Tasks	Reason For Difference
Scheduled Sub-Task Attributes	Actual Sub-Task Attributes	Reason For Difference
Scheduled Resources	Actual Resources	Reason For Difference
Forecast Environment (Conditions)	Actual Environment (Conditions)	Reason for Difference

Appendix B

Identification of Edge-Matched Maximal Isomorphic Subgraphs in Distinct Graphs

- 1) Enter and store labelled graphs (labels stating relations $\{r_{ij}\}$ link between nodes $n_i, i=1, \dots, n$); nodes are entered as $\{n_i | i=1, \dots, n\}$, edges are entered as $\{r_{ij} | r_{ij} = \text{relational label}, r_{ij} = 0 \text{ if there doesn't exist an edge between } n_i \text{ and } n_j\}$
- 2) Store graphs as $\{(\{nodes\} n_i, \{rij\})\}$
- 3) Enter new graph (target) and identify all subgraphs
- 4) Identify all stored graphs (probes) with non-trivial (at 2 edges in common) subgraphs isomorphic to a subgraph/subgraphs in target. Record these non-trivial isomorphic subgraph pairings
- 5) Reject isomorphic subgraphs in which corresponding edges $\{(l,j)\}$ (compare subgraph with isomorphic target subgraph) have non-identical corresponding edge labels $\{r_{ij}\}$
- 6) Order remaining subgraphs in terms of decreasing $|\{\text{corresponding identically labelled } (\{r_{ij}\}) \text{ edge sets}\}|$
- 7) Identify corresponding trees with maximal $|\{\text{corresponding identically labelled } (\{r_{ij}\}) \text{ edge sets}\}|$

Maximal Subgraph Matching Algorithm with Extension to Analogical Inference:

Consider the relational structure of Target $T\{t_i | i=1, \dots, T\}$

Compare Target T with all stored graphs G_k (potential probes) $k=1, \dots, g$; $g =$ number of stored graphs

(*1) Identify which G_k contain a connected subgraph SG_k with relations identical to an isomorphic (*2) subgraph T' of T .

Identify those SG_k with maximum number n^* of nodes

They form a set $S_{n^*} = \{SG_m | m=1, \dots, g'\}$

Extend each SG_m back to G_m (not necessarily isomorphic to T)

These G_m can be compared with T to identify edges (p,q) in G_m but not in T .
Where p, q range over the nodes of SG_m

The relations r_{pq} corresponding to these edges (p,q) can be tested (in the context /environment of T) to determine if they hold true/are relevant with respect to T .

If so, then an inference has been made in T using analogical reasoning with respect to the G_m .

If this leads to no (suggested inferences) then repeat with next set of maximal subgraphs ($\#nodes=n*-1$)...and so on

(*1)

Identify which G_k contains a subgraph SG_k with relations identical to a subgraph T' of T

Compare $\{r_{ij}(T')\}$ with isomorphically corresponding $\{r_{ij}(SG_k)\}$

NB this may require relabelling the nodes of SG_k so that isomorphically corresponding nodes bear the same numerical labels

Identify (i,j) for which $r_{ij}(T')=r_{ij}(SG_k)$

Determine if these corresponding edges form a connected (sub)graph, SG_k :

- i) Identify i_{min} and j_{max} from $\{(i,j) \mid r_{ij}(T) = r_{ij}(G_k)\}$
- ii) For $i_{min} + 1, \dots, i, \dots, j_{max} - 1$, check if there exists an i that occurs as either a first index of an edge or a last index of an edge but not both
- iii) If such an i exists then SG_k is rejected.

(*2)

To identify if there exists an isomorphic correspondence between subgraph T' and SG_k where $n' = \#nodes(T') = \#nodes(SG_k) = k'$:

There are $(n')!$ 1:1 maps from T' to SG_k

Identify those which also form 1:1 maps from the edge set of T' to the edge set of prospective S_k , which preserve the degree at each node of T' ignoring edges in T but not T' , and SG_k , ignoring nodes in G_k but not S_k

To identify all the 1:1 maps from the edge set of T' to the edge set of prospective S_k , which preserve the degree at each node of T' ignoring edges in T but not T' , and SG_k , ignoring nodes in G_k but not S_k :

Choose any node w of T'

Map to any node w' of S_k

Choose a node x of T' such that (w,x) is an edge of T'

Map x to a node x' of S_k such that (w',x') is an edge of S_k

If $\text{degree}(w) > 1$ then repeat the above with a different node w'' of S_k

Repeat until the number of (mapped edges) from w in $T' = \text{degree}(w)$ in T

Choose another node in T' other than w and repeat the above

Continue until (if possible) all nodes and edges in T' are mapped 1:1 to nodes in SG_k

If this is achieved then T' is isomorphic to S_k