Classification Networks: A Knowledge Representation Scheme for Curriculum Prescription

Kwok-Keung Yum
Centre for Computer Based Training
and Decision Support Systems
University of Western Australia
Nedland, WA 6009, Australia

Thomas J. Richards
Department of Computer Science
and Computer Engineering
La Trobe University
Bundoora, Vic 3083, Australia

Abstract

This paper presents a classification scheme that articulates the categorisation of subject matter. Two basic classificatory devices are introduced: (a) a feature representing the resulting category characterised by a property that the feature denotes, (b) a dimension representing a perspective that partitions the Universe of Discourse (UoD) or a category.

A classification based on a dimension can be further classified into the categories it has formed. Dimensions can be juxtaposed to form a classification based on multiple perspectives. The classificatory devices of features and dimensions, which form a classification network, support a wide range of subject organisation types, viz. (a) conceptual organisation, (b) procedural organisation and (c) theoretical organisation. The purpose of the approach is to support the clarity, simplicity and maintainability of large scale general ITS development.

Some simple course organisation examples using classification networks are also presented.

1. Background

In the intelligent tutoring systems (ITS) community, there have been very few discussions about methodological issues such as the kinds of knowledge that can be used for what design purposes. In particular, when dealing with larger scale ITS developments, there is a tendency to confuse two different design processes: (1) prescribing the curriculum (what *should* be there for pedagogical purposes?) and (2) representing content knowledge (what should be there for the running of the tutoring system?). Curriculum prescription is a human-based design process, specifying what is important and which material is more important than the other. The (partial) ordering of the subject categories is subject to the emphases of the course and the student background. The representation of content knowledge is also a design process, but it is based on both human understanding and machine efficiency (epistemological adequacy). From the ITS designer's point of view, curriculum prescription and content knowledge representation are two distinct processes serving different purposes and they should be clearly separated.

The possible confusion mainly stems from one major source, i.e., there are no separate knowledge representation schemes for these two important processes. Curriculum prescription is usually regarded as a part of the knowledge representation. A classical example is Carbonell's SCHOLAR where the South American geographical knowledge was structured using the semantic network formalism. There were some elements of curriculum organisation in SCHOLAR (e.g. South America has geographical features comprising rivers, mountains, etc.), but the knowledge was primarily structured to respond to the student's questions. Along this line of thinking, the knowledge structuring of subject matter in conventional ITSs needed to take into consideration three major factors: (1) the student's initiative, (2) the subject matter and (3) the machine's coding. The indiscriminate consideration of diversified factors tended to be an impediment to large scale ITS developments.

In the mid-eighties, Bonar *et al.* [1986] proposed an object-oriented organisational approach, representing pedagogical *issues* [Burton & Brown, 1982] and tutoring functionalities as classes. Tutoring functionalities formed the superclasses of the curriculum issues. In one respect this organisational approach widened the dimension of manipulation for curriculum structuring. The relations between curriculum objects were reduced to the partial class-subclass ordering, which gave the ITS developer a much simplified handle for curriculum prescription.

Lesgold [1988] pointed out that a curriculum structure should serve the following educational purposes: (1) ordering subject content for teaching, (2) providing coherence that "glues" together content knowledge. Conventional curriculum structures emphasised the presentation order of subject matter, but they generally failed to specify clearly the "glue" connecting subject contents. In conventional ITSs, the coherence between content knowledge was neither taught nor used to diagnose the student errors.

The major concern of this paper is the *formalisaiion* of the *curriculum structure* in terms of lesson *features* (major pedagogical issues). The curriculum designer specifies the features and feature relations that categorise subject matter. An automatic program that traverses the network of features and feature relations (called the *classification network)* in a top-down manner is then applied to collect the subject categories (called topics) for teaching. Since each topic is characterised by some combination of features (called index feature pattern), the teaching of a topic may then include the teaching of the index feature pattern. The index feature pattern is the "glue" that connects the topics.

2. The epistemological hierarchy of the subject matter

We may follow Reigeluth and Stein's [1983] "zoom lens" metaphor to illustrate the significance of the organisation of the subject matter. Studying a subject is similar to viewing a scene with a movie camera. One starts with a wide-angle view which allows one to sec the major ideas but without any detail. Then one zooms in on a part to get a finer view. Zooming-in can be selected "downward" (one level of finer detail) or "across" (jumping from one idea to another, but keeping the level unchanged). The earlier ideas are those that are crucial for the later development and they have to be presented at an earlier stage of teaching so that the later presentations can depend on them. Following this metaphor, one may see that a subject may consist of many top-level ideas. Each idea may be refined into many second level ideas and so on. These ideas depend on certain perspectives from which the subject matter is "viewed". Based on the general type of subject knowledge, three types of organisation structures are possible [Reigeluth & Stein, 1983J:

(1) conceptual organisation shows superordinate/ coordinate/subordinate relationships among concepts; subsuming concepts are always near the top level of the knowledge hierarchy;

(2) procedural organisation shows relationships between steps of a procedure; the order may be derived from procedural steps or decisions about procedural applications;

(3) theoretical organisation shows change relationships among principles; the order may be derived from the chains of interrelated descriptive principles or the chains of prescriptive principles that optimise or influence some desired outcome.

3. Organising curriculum using features

Subject knowledge can be considered at two levels: (1) instance level and (2) category level. Instances correspond to factual knowledge and categories correspond to conceptual knowledge, procedural knowledge or principles. The set of all instances forms our Universe of Discourse (UoD). Subject categories for learning are not decided arbitrarily. On the one hand, they must be compatible with the nature of the subject and the student understanding; on the other hand, they must conform to some obvious, important categorisation conventions:

Related categories are formed from some perspective that divides and groups the items for learning.[1] For example, in the subject of set enumeration, the perspective of "set representation" differentiates between two categories: "member-listing" and "property-listing". There are only a few categories formed in a perspective but there must be at least two such categories.[2] All categories in a perspective must be non-empty. Not all items in the UoD can be categorised in a perspective, but those items that can be categorised in a perspective must be exhaustively classified. All categories formed under a perspective must be mutually disjoint (i.e. there is no common item shared between any two categories).

Organising curriculum is a process of categorising subject knowledge in a variety of perspectives. By representing categories and perspective as some linguistic elements, we may formalise the subject categorisation as follows:

Feature. A feature denotes a property that is possessed by all knowledge instances in a category. Thus a feature is a name referring to a property as well as to a category. For example, the feature member-listing refers to the category "member-listing" mentioned above.

Feature pattern. Features represent only primary categories. Sometimes we need to consider complex categories such as the union and the intersection of categories. A union of categories is represented by (or featurei ... featuren), and an intersection of categories is represented by (and featurei featuren), where feature! denotes some (primary) categories, (l<i<n, l<n). A logical expression of features like these arc called feature patterns. Feature patterns of and- and or- types can be further combined to denote the union of intersections, and so on.

Dimension. The categorisation in a perspective is formalised as a dimension, a data structure consisting of a list of features denoting the categories that are classified in a perspective. Such features are called the differentiation features of the dimension. There are only a few differentiation features in a dimension, but there must be at least two such features. All categories named by features in a dimension must be non-empty. Not all items in the UoD can be categorised by the differentiation features in a dimension, but those items that can be categorised by the differentiation features of a dimension must be exhaustively classified. All categories formed by the differentiation features must be mutually disjoint. Although it is not absolutely necessary, dimensions are conveniently referred to by names. Graphically, the differentiation features are listed top-down in a column. They are put next to a left bracket

1 Perspective is the ability to think clearly and sensibly about a situation and consider it in relation to everything else [Collins COBUILD English Language Dictionary, 1990].

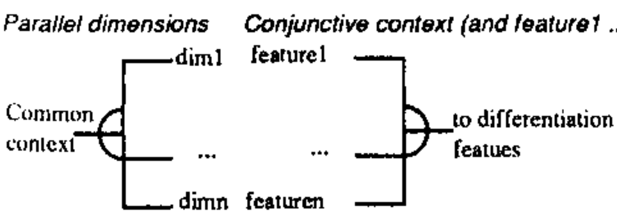
^ A perspective is a view for categorising and differentiating. If there is only one category grouped in a perspective, there is no need for that perspective.

E The number of "teeth" in the bracket should match the number of differentiation features. For example, in a subject matter related to the teaching of electricity, a curriculum developer may find it appropriate to consider the subject matter organisation according the dimension LAW. Under the dimension LAW, one may want to introduce two basic laws, Ohm's Law and Kirchoffs Law, and contrast them with each other. In that case, the differentiation features {OhmsLaw, KirchoffsLaw} is good for the purpose. Since the subject matter can be viewed from different perspectives, one may consider *parallel* dimensions, e.g., (a) Law, (b) ProblemType, each of which has its own distinct differentiation features (Figure 1).

Context. Each dimension has a part (called *context)* denoting the category to which the categorisation perspective applies. There are two types of dimensions: (a) primary, (b) non-primary. A *primary dimension* represents the classification that is carried out on the whole UoD, not on any category. In that case, we attribute a pseudo-feature Nep (Network entry point) as the context. For example the dimensions Law and ProblemType do not have any context.

A *non-primary dimension* represents the classification that is carried out on a proper category of the UoD. It's context is a feature pattern representing the category to which the classification applies. For example, in the above electricity teaching scenario, one may consider the dimension RI-VI for the teaching of the qualitative relationships: resistance-current and voltage-current. Such differentiations of feature can only happen to those subject items categorised under the features *OhmsLavt and* QualProb (i.e. the qualitative nature of Ohm's Law). In this case, one has to include in the dimension, i.e. RI-VI, the context (and OhmsLaw QualProb), which represents a category grouped under *both* features OhmsLaw and QualProb. The dimension RI-VI offers two differentiation features {RI, VI} (Figure 1).

Finally we summarise a graphical notation for the representation of parallel dimensions and contexts as follows. Parallel dimensions having the same context are joined together by a conjunctive joint-platform. A conjunctive context has a conjunctive receiving-platform:



Dominance. A feature A is *dominant over* a feature B if the feature A exhibits whenever the feature B exhibits. For example, the feature basic-wave-behaviour is dominant over the feature light-behaviour because the feature basic-wave-behaviour exhibits whenever the feature exhibits light-behaviour. A dominating feature must be a feature mentioned in the context of a dimension; it dominates all differentiation features of that dimension and all features in the dimensions that follows it.

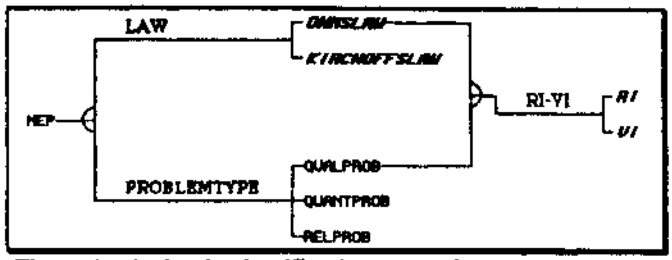


Figure 1: A simple classification network.

4. Instances and characterisations

From now on one may consider a declarative approach of forming basic categories by exhaustively and exclusively *listing* all features possessed by all instances in that category. Given a classification network, a *basic category* is a non-empty category that can be formed by listing in a set *all* features which *every* instance in that category exhibits, but not features that *some* instances in that category do not exhibit.[3] The set of all such features is called a *characterisation* of that basic category. More precisely, if *every* instance in a basic category exhibits a feature, then the characterisation of the category *consists* of that feature; otherwise, the characterisation of the category *does not consist* of that feature. Referring back to the electricity lesson in Figure 1, the features arc considered from {OhmsLaw, KirchoffsLaw, QualProb, QuantProb, RelProb, RI, vI). The category represented by (and OhmsLaw QualProb RI) is a basic category because all features possessed by all instances in the categories are listed in the set lOhmsLaw, QualProb, RI), while all other features, viz. KirchoffsLaw, QuantProb, RelProb, and VI are not features possessed by the instances in that categories. The set {OhmsLaw, QualProb, RI} is the characterisation of a basic category.

The notion of characterisations provides two important properties that helps the organisation of curriculum:

(1) The collection of all basic categories associated with characterisations form the finest classification of the UoD.[4] For example, the problem "Given that the voltage across A & B is 10 volts, if the resistance between A & B is increased by two times, how does the current between A & B change?" is an instance problem from the characterisation (OhmsLaw, QualProb, RI). All other categories can be described as unions of basic categories.

[3] Not all categories are basic categories. For example, the category (and OhmsLaw QualProb) is not a basic category.

[4] If there is a finer category, it must exhibit some extra features, but this is impossible as it has been assumed in the definition of a classification that *all* features have been included.

(2) The syntax of classification networks provides constraints on the occurrence of features in characterisations. One may define a feasible feature combination as a set of features that is collectively possessed by all instances in a category of the UoD, and a maximal feasible feature combination as a feasible feature combination A such that for any feasible feature combination $X, X \subseteq A$. The partial ordering of subsets restricted to the set of all feasible feature combinations has a one-to-one correspondence with the partial ordering of supersets restricted to the set of all corresponding categories in the UoD. That is, if A and B are feasible feature combinations, and A' and B' are their corresponding categories, then $A \supset B$ if and only if $\mathcal{A}' \subset \mathcal{B}'$.

A characterisation is a maximal feasible feature combination: it is a feasible combination because all constraints on features dimensions in the classification have been satisfied in the characterisation; it is maximal because all features that occur in the classification network are exhaustively and exclusively listed. While a characterisation has been defined semantically as a reference to a basic category, its property of being a maximal feasible feature combination allows us to consider the syntactic constraints of features imposed by the classification network (availability of features). Based on such constraints, one may build tools (the classification network editor) to help the input, the consistence checking of the classification network, and the generation of characterisations. Further detail about the mathematical foundations of the classification network and the building of the classification network editor can be found in [Yum, 1993].

When using the classification network to categorise a subject, only features and characterisations are of operational significance. That is, classification operates only at the syntactic level of of features and characterisations, not at the semantic level of categories and instances. Knowledge categories that are useful for teaching are represented as a general tutorial structure of topics. Specific content knowledge such as tasks and problem instances is separately represented. Tasks and problem instances are indexed into the topic hierarchy using two keys: (1) their names, and (2) their context of applications in terms of feature patterns.

5. Forming topics

A topic is a generic unit tutorial structure which contains materials for instruction. All topics together form a lesson. Topics and the lesson are the only curriculum structures considered in this paper. Topic structuring is a process of creating topics for presentation.

Topics in this paper are categories characterised by the ordering of features and dimensions in a classification network. A classification network makes explicit the following classificatory criteria:

(1) Dominance: simple ideas form dominating features which are the context of some classificatory dimensions.

(2) Differentiation: refined ideas form the differentiation features of some dimensions.

(3) Default ordering of differentiation features and parallel dimensions: differentiation features and parallel dimensions are arranged in ascending order of difficulty.

(4) Key features: not all features included in the classification network are important or significant enough to deserve the opening of a topic for presentation. Features that are worth teaching are marked as key features. Non-key features serve only as the conceptual scaffolding used to link up the key features to be introduced later. Pictorially, a key feature is typed italic bold font, as distinct from the non-key features which are in plain font. In Figure 1, the differentiation features in the dimension ProblemType are all non-key features. All other features are key features.

The basic idea of topic structuring is to preserve the hierarchical ordering of the key features already made explicit in the classification network. This requirement restricts the topic collection algorithm to two basic alternatives: a depth-first search or a breadth-first search on the network. Topics are subject categories that are characterised by the index feature pattern collected in the following manner:

> Nep forms the topmost topic of the topic hierarchy.
> Dominating key features form dominating topics and dominated key features form dominated topics.
> Differentiation key features of a single dimension are used to form "parallel" topics at the same level.
> Key features from parallel dimensions are grouped to form a topic.

Each topic is represented as an object (in the sense of object-oriented programming) having the slots: (1) index-feature-pattern (a feature pattern that characterises the topic), (2) breakdown-index-feature-pattern (a feature pattern for further breakdown), (3) features-to-teach (features that should be taught in the topic), (4) dimensions-to-teach (dimensions that should be taught in the topic), (5) characterisations (the collection of all characterisations that should be covered in that topic), (6) subordinate-topics (topics are subsumed under this topic) and (7) superordinate-topics (the topic that subsumes this topic). There are some other additional slots for housekeeping purposes, but these are not the major concern of this paper.

Figure 2 shows the topic hierarchy generated from a depth-first traversal using the classification network in Figure 1. The reader may notice that the ordering of Topics 2 and 3 are is the reverse of that of the differentiation features OhmsLaw and KirchoffsLaw. This is because the ordering of differentiation features takes into account the number of key features that may follow these differentiation features. In this particular case, the number of key features that follow KirchoffsLaw is 0, and the number of key features that follow OhmsLaw is 1. Therefore, the topic with fewer key features to follow (i.e. Topic-2) is generated first. In real teaching situations, Kirchoff s Law is harder than Ohm's Law and there should be more key features to follow than in Ohm's Law. The ordering of Topics 2 and 3 should not happen.

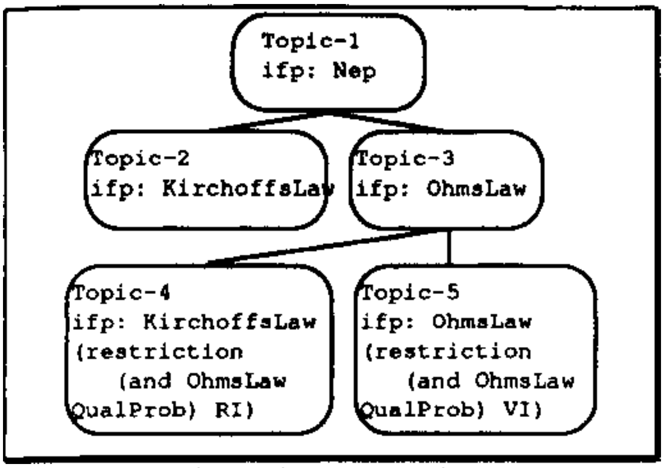


Figure 2 Topics formed from the classification network in Figure 1, where 'ifp' stands for 'index feature pattern'.

6. Using the classification network

It has been a major theme in the ITS community that ITSs should be driven by "intelligence". In order to show the intelligence', the development of curriculum structure is also associated with specific tasks (e.g. WUSOR [Goldstein, 1982], EUROHELP [Breuker, 1990]). This view opens up a possible confusion between knowledge structuring for pedagogical purposes and content knowledge representation for designing the "intelligence" part of the system. In order to structure knowledge for pedagogical delivery, one has to consider the task specifics at the very beginning of ITS development. This is a major obstacle on the path towards large scale ITS development.

Practical teaching shows otherwise. Experienced teachers need only skeletal guidance from the curriculum; they do not need detailed examination of subject content in order to "prepare" the lesson. They know the detailed content knowledge very well beforehand. In lesson preparation, they organise only the presentation skeleton of the subject.

Based on this observation, we divide ITS design into three major phases. (1) Designing generic tutorial strategies in terms of generic tutorial structures like the lesson, topics, tasks, problem instance generators and task steps. (2) Specifying the curriculum structure (the topic hierarchy) in terms of features and feature relations. (3) Representing content knowledge (tasks and instance generators) and indexing the content knowledge elements according to their names and their contexts of application. These three phases are primarily independent of each other. The reader may find a high level description of these three phases in [Yum & Richards, 1992a, 1992b]. Further detail can be found in [Yum, 1993].

The approach of feature based curriculum structuring has a deep implication in ITS development. Apart from formalising the curriculum structure, features can be used in lesson delivery and student diagnosis as follows:

indexing subject tasks by feature patterns that represent the contexts of application of the tasks;
teaching the feature patterns that characterise the topics;
teaching the contexts of task application;
- diagnosing student performance in terms of features.

Indexing tasks using features is a key to the link-up of the curriculum module and subject content knowledge. Since topics are connected by feature relations, one can design tutorial strategies to remind the student of those learned features that are most relevant to the current learning. At present, the tutorial objectives do not include any consideration of teaching the contexts of task application or diagnosing student performance.[5]

7. Related work

BIP (for BASIC Instructional Program) organised its curriculum information network at three levels [Barr *et al.*, 1976]. At the top were technique units which must be ordered in advance according to the prerequisite relation. At the middle level were skill units which were linked to the upper level of techniques and to the lower level of tasks that exercised the skills. BIP's student model was defined in terms of skills. Problems are selected to exercise the greatest number of skills without including any skills beyond the student's reach.

BIP-II [Wescourt *et al.* 1977] refined the organisation of the skill units in BIP's curriculum information network. The skills required for one technique were connected by links representing a variety of relations like subtype relations, whole-part relations, functional dependence, difficulty levels, etc. Since the skill elements had been made explicit, the construction of the skill sets in BIP-II was more precise than that in BIP. The skill elements of BIP-II were used only for lesson delivery; the use of these elements in student diagnosis was only mentioned

Alan Lesgold [1988] formulated a more refined curriculum structure, which has three layers of knowledge. The bottommost is the *knowledge layer* that contains the representation of the knowledge that the system is trying to teach. This layer corresponds to the tasks in this paper. The middle is the *goal lattice layer* that contains the "learning goals" which describe various kinds of learning associated with the knowledge layer. It is the combination of these "learning goals" that gives the tutor a sense of coherence in teaching, and allows the tutor to determine the similarities and differences of the present and the previous learning. By and large, this layer corresponds to the feature relations presented in a classification network. The topmost layer is the *metaissue layer,* which contains a number of

[5] Since tutorial steps in this paper are represented as procedures, it is possible to specify procedures to teach the context of task application or diagnosing student performance. However, these considerations are outside the scope of the current research.

"viewpoints" grouping together various learning goals to form a lesson. Roughly the metaissue layer corresponds to our topic hierarchy. Like BIP-IFs curriculum structure, Lesgold's metaissues, goals and knowledge layers are interconnected explicitly by the developers. Unlike the curriculum structure of BIP-II, which is mainly used for lesson delivery, Lesgold's curriculum structure is used to teach the lesson as well as to do student diagnosis.

8. Summary

We have presented a classification scheme that makes explicit the curriculum skeleton. The major emphases of the classification network are: (1) supporting human-oriented design (focusing on educational consideration rather than on machine performance); (2) separating the often confused conceptions of curriculum design and content knowledge representation; (3) enabling the tutorial strategies to include the teaching of topic coherence; (4) allowing specific knowledge to be indexed into the curriculum skeleton; (6) logically separating the design issues of "What is it to be used?" (classification networks and topics hierarchies) "Who is using?" (instructional objectives) and "How is to be used?" (specific knowledge and methods). The purpose of the approach is to support the clarity, simplicity and maintainability of large scale ITS delivery.

At present, we have implemented a small ITS based on Reigeluth's Elaboration Theory of Instruction [Reigeluth & Stein, 1983]. The subject of the ITS was a procedural domain of elementary set theories. Topics were organised using the classification network. Implemented tutorial strategies included: (1) topic orientation, (2) teaching features that characterise the topic, (3) presenting individual tasks in an expository manner [Merrill, 1983]. We expected to use two student models to capture the dynamics of the teaching. The first student model was designed for the diagnosis of learning a single task. The current implementation used only a simple overlay student model. The second student model was intended to be used to diagnose inter-topic transition in learning. To date, we have not yet started on the second student model.

References

[Barr et al., 1976] Barr, A., Beard, M. & Atkinson, R.C 1976, The computer as a tutorial laboratory: the Stanford BIP Project', International Journal of Man-Machine Studies, vol. 8, pp. 567-596.

[Bonar et al., 1986] Bonar, J., Cunningham, R. & Schultz, J. 1986, 'An object-oriented architecture for intelligent tutoring systems', Proceedings of the ACM Conference on Object-Oriented Programming Systems, Languages and Applications, pp. 269-276.

[Breuker, 1990] Breuker, J. (ed.) 1990, EUROHELP: Developing Intelligent Help Systems, EC, Kopenhagen.

[Brown & Burton, 1982] Brown, R.R. & Burton, J.S. 1982, 'An investigation of computer coaching for informal learning activities', in Intelligent Tutoring Systems, eds D. Sleeman & J.S. Brown, Academic Press, London, pp. 79-98.

[Goldstein, 1982] Goldstein, LP. 1982, The genetic graph: a representation for the evolution of procedural knowledge', in Intelligent Tutoring Systems, eds D. Sleeman & J.S, Brown, Academic Press, London, pp. 51-77.

[Lesgold, 1988] Lesgold, A. 1988, Toward a theory of curriculum for use in designing intelligent instruction systems', in Learning Issues for Intelligent Systems, eds H. Mandl & A. Lesgold, Springer Verlag, New York, pp. 114-137.

[Reigeluth & Stein, 1983] Reigeluth, C.M. and Stein, F.S. 1983, "The elaboration theory of instruction.", in Instructional-design Theories and Models: An Overview of Their Current Status, ed. CM. Reigeluth, Lawrence Erlbaum Associates, Hillsdale, New Jersey, pp. 335-382.

[Merrill, 1983] Merrill, M.D. 1983, 'Component display theory', in Instructional-design Theories and Models: An Overview of Their Current Status, ed. CM. Reigeluth, Lawrence Erlbaum Associates, Hillsdale, New Jersey, pp. 279-333.

[Wescourt et al, 1977] Wescourt, K., Beard, M. & Gould, L. 1977, 'Knowledge-based representation sequencing for CAI: application of a network representation, quoted in Wenger, E. 1987, Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge, Morgan Kaufmann Publishers, Los Altos, California, pp. 110-111.

[Yum & Richards, 1992a] Yum, K.K. & Richards, T.J. 1992a, 'Instruction as reasoning about multiple objectives', in Intelligent Tutoring Systems: Second International Conference on Intelligent Tutoring Systems, ITS'92, Montreal, June 1992, Proceedings, eds Frasson, C, Gauthicr, G. & McCalla, G.I, Springer Verlag, Berlin, pp. 199-208.

[Yum & Richards, 1992b] Yum, K.K. & Richards, T.J. 1992b, 'Simulating teaching by reasoning about instructional objectives', Proceedings of the Third Annual Conference on AI, Simulation and Planning in High Autonomy Systems (AIS'92), Perth, Australia, July 8-10, 1992, IEEE Computer Society Press, Washington, pp. 86-92.

[Yum, 1993] Yum, K.K. 1993, Reasoning about Multiple Instructional Objectives: A Framework for Building Intelligent Tutoring Systems through Abstraction, Ph.D. Thesis, Department of Computer Science and Computer Engineering, La Trobe University.