Defining "Ill-Defined Domains"; A literature survey.

Collin F. Lynch
& Kevin D. Ashley
Intelligent Systems Program,
University of Pittsburgh.
Pittsburgh, Pennsylvania.
(collinl@cs.pitt.edu)
(ashley@pitt.edu)

Vincent Aleven
& Niels Pinkwart
Human Computer Interaction Institute,
Carnegie Mellon University.
Pittsburgh, Pennsylvania.
(aleven@cs.cmu.edu)
(nielsp@cs.cmu.edu)

Abstract: In order to make progress on Intelligent Tutoring in ill-defined domains it is helpful to start with a definition. In this paper we consider the existing definitions and select one for the basis of our discussion. We then summarize some of the more salient characteristics of ill-defined domains from a tutoring standpoint and some human and ITS strategies that have been employed to cope with them. We conclude with some challenges to the ITS community to spur further research.

Keywords: ITS 2006 workshops, intelligent tutoring systems, ill-structured domains, ill-defined domains

INTRO

Intelligent Tutoring Systems (ITS) have made great strides in recent years. Many of these gains have been made in well-defined domains such as geometry, Newtonian Mechanics, and system maintenance. In recent years similar gains have been made in ill-defined domains such as law, design, and composition.

In this paper we present an overview of ITSs for ill-defined domains. Our goal is to provide a framework for discussion and research in this area by synthesizing past efforts. We begin by providing a working definition of the term. We then highlight the relevant characteristics of ill-defined domains as they pertain to tutoring. Following this, we summarize past research in this area in terms of some of the human and ITS tutoring strategies that have proven successful.

In order to frame our discussion we will illustrate our points using the following domains:

Physics: What is the airspeed velocity of an unladen swallow? [1]

Ethics: Is it morally justified to raise swallows for food?

Law: If I raise swallows on my property am I liable for damage to my neighbors' property?

Architecture: Design a residential building with housing for swallows.

Music Composition: Write a fugue based upon a swallow's song.

This is not an exclusive list. Rather it is a representative sample meant to motivate further discussion.

For the purposes of this workshop we have chosen the term *ill-defined domain*. The terms *ill-structured* and *ill-defined* are used interchangeably in the literature. To avoid confusion we will only use the former. Much of AI and education literature is framed in terms of *problems* rather than *domains*. *Problem* typically connotes achieving a specific goal in a concrete scenario using methods amenable to state-space search. *Domain* typically connotes an area of study such as physics or a set of problems. For the purposes of our analysis, this distinction is immaterial. We have chosen *'domain'* in order to emphasize that the end goal of tutoring is typically general domain knowledge or problem solving skills, not problem-specific answers. At the same time, we recognize that domains like physics have both well-defined and ill-defined subdomains.

[1] See: "Monty Python and the Holy Grail" (1975) *Python et al.*

DEFINITION

In order to have a serious discussion of ill-defined domains it is necessary to define the term. This will be problematic as the term has been given a wide variety of definitions in the literature. Much of the literature on this subject grows out of either AI, or Decision-Making under Uncertainty. We will begin by discussing the historical AI-centric definition followed by a discussion of competing theories.

Ill-defined domains have a long history in AI. The earliest work by John McCarthy is referenced in Marvin Minsky's 1961 paper "Steps Toward Artificial Intelligence" [Minsky, 1995]. According to McCarthy and Minsky, a *well-defined domain* is one in which there exists a systematic way to determine when a proposed solution is acceptable. *Ill-defined domains*, by definition, lack such a procedure. This residual definition underpins most of the subsequent work in this area.

Four subsequent definitions have influenced our analysis. They vary in style and meaning based upon the questions that their authors were asked. Reitman [Reitman, 1964] followed Minsky's definition but sought to impose some structure on the implicit concept of ill-definedness to facilitate more serious research. His work was based upon the composition of fugues by an expert composer. Among other things, he noted that the sole requirement of this task was that the result "be a fugue."

Newell [Newell, 1969] asked why a given domain might appear ill-defined to one problem solver and well-defined to another. His definition was framed in terms of a solver's ability to identify a "specific" answer. Simon [Simon, 1973] by contrast sought to identify why ill-defined domains were *not* amenable to state-space search.

Simon's discussion was built upon his past work with the General Problem Solver [Newell and Simon, 1995]. He used this framework to consider classically ill-defined domains such as architecture. In this process he highlighted some of the salient characteristics of such domains including the lack of a clean decomposition and the relationship of scope to definedness. In chess, for example, the selection of a single move may be well-defined while winning an entire game is not. This analysis led to similar work in other domains such as design [Goel and Pirolli, 1992] and medicine [Pople, 1982].

This line of reasoning has found its way into the psychological literature. Voss and Post's [Voss and Post, 1988], paper considers several varying definitions of ill-definedness. These include the work of Johnson [Johnson, 1988], and Lawrence [Lawrence, 1988]. Johnsons' work drew on notions of ill-definedness in the expert problem-solving literature while Lawrence chose the domain of law. Voss and Post based their working definition on Reitman's [Reitman, 1964]. It focused on the "open-constraints" present in ill-defined problems and domains. They specifically emphasized the constraint-propagation aspects of ill-defined problem solving. We will return to this aspect below.

Most recently, Ashley and Pinkus [Ashley et al., 2004] defined ill-defined domains as having the following key characteristics: 1) they lack a definitive answer; 2) the answer is heavily dependent upon the problem's conception; and 3) problem solving requires both retrieving relevant concepts and mapping them to the task at hand. In identifying these characteristics, their goal was to motivate the development of an ITS for applied ethics. Their analysis, like ours, is driven by a set of indicative examples.

Our goal in this paper is to present a summary of ITS research in ill-defined domains with a focus on the characteristics of those domains that affect ITS design. For this reason we have chosen to follow Ashley and Pinkus' methodology in this paper.

RELEVANT CHARACTERISTICS

In this section we summarize five key characteristics of ill-defined domains that have been discussed in the literature. We give an intuitive description of each one highlighted by examples from the sample domains. This list is not intended to be exhaustive but to serve as a basis for subsequent discussion.

3.1 Verifiability

"What is the airspeed velocity of an unladen swallow?" This is a classical physics problem (according to Monty Python) that could be solved by applying the theory of Newtonian Mechanics. The answer can be calculated programmatically and verified empirically to an arbitrary degree of precision. This is not the case with domains such as law. Legal arguments may be judged functionally (win or lose) or aesthetically (good or bad) but no unambiguous standard exists. While there are valid arguments for or against some solutions there often is no one *right* answer.

Domains like architecture and music are even less verifiable. While arguments may be made for or against a given instance, such arguments are necessarily qualitative. Brolin's argument against modern architecture [Brolin, 1976], while convincing, is based entirely on aesthetic value judgments, not on any absolute or quantitative measurements. Ultimately, in such domains, one man's masterpiece is another man's trash.

3.2 Formal Theories

Valid formal theories such as Newtonian Mechanics provide a means to determine a problem's outcome and test its validity. A formal theory in physics is considered valid if its predictions can be verified empirically, that is, it accurately describes all relevant phenomena. Physicists are engaged in a continuous process of theory formation. Their ultimate goal is to develop a single cohesive theory that explains all physical phenomena.

This process becomes difficult in new regions of physics (e.g., astrophysics) where empirical verification may be untenable. A number of competing theories may coexist and, while they stem from testable phenomena, they may not be readily falsified. In the realm of scientific discovery, argumentation may involve a degree of interpretation that approaches that in domains like law.

Lawyers, like physicists, are also engaged in a continuous process of domain structuring ([Levi, 1949], [Radin, 1933], [Schauer, 1998]). The goal however, is typically prescriptive, not descriptive ([Schauer, 1998], [Llewellyn, 1981]). More formal legal theories are typically based on statutes or case decisions and exist to prescribe what *should* be done in a specific case, not necessarily what *is*. Such theories are relatively specific and may not be expected to generalize across legal fields. While such prescriptions are normatively based, the norms may change over time. Formal attempts to model the structure of the law are usually in flux [Lehmann et al., 2005].

Physicists seek out formal theories. Lawyers invent such theories as needed but acknowledge their limitations. Architects by contrast typically shun such theories as being overly restrictive. While such theories may be accepted (e.g. [Alexander, 1977], [Alexander, 1979]), they are typically used to guide intuitions, not to dictate results [Goel and Pirolli, 1992].

3.3 Task Structure

Physics is largely a descriptive domain. Most textbook physics problems are similar to the swallow problem above. Given some information, compute a desired quantity using a formal theory. Research physicists, by contrast, seek to formulate new theories that explain observed phenomena or to observe phenomena that may be used to falsify existing theories. Both tasks involve elements of design and as such are necessarily ill-defined.

Law is both an analytical and design domain. Legal analysis includes determining what laws or theories are applicable to the current situation and what result they would prescribe ([Sergot et al., 1986], [Radin, 1933], [Schauer, 1998]). Such tasks are necessarily ill-defined much like medical diagnostics [Pople, 1982]. Legal design tasks include the formation of arguments that would necessitate or evade such analysis to achieve their desired goal. The study of architecture, by contrast, is characterized primarily by design tasks. In such cases novelty, not repetition, is the goal. While formal theories might be used to teach or guide intuitions, practitioners typically shun such programmatic analyses.

3.4 Open-Textured Concepts

Open-textured concepts are abstract concepts such as "vehicle" and "space" that have an inherent indeterminacy [Gardner, 1987] and lack any absolute definition. Such concepts are a defining characteristic of legal theories ([Ashley, 1990], [Berman and Hafner, 1985], [Sergot et al., 1986]) and architectural theories like the Pattern Language. They become problematic when they must be applied to concrete elements.

In many ways this is also true of physics. While physicists seek to describe real-world phenomena, physical theories still make use of concepts such as "time points" and "energy". While these concepts are defined within the physical theories, their applicability to new phenomena (e.g., black holes) is often a matter of debate. As with law and architecture, the application of a theory depends on the definition of its terms. This issue has been discussed by [Reitman, 1964], [Simon, 1973], and [Ashley et al., 2004].

Computational models of these domains typically handle open-textured concepts in one of four ways: impose a definition arbitrarily; require that the user provide one; reason from case examples; or ignore the issue entirely. In the last option, they are focusing on the theory alone. Each alternative constrains the system in some way either to a limited well-defined domain or an ad-hoc set of cases.

3.5 Overlapping Subproblems

Human and non-human problem solvers typically decompose a given problem into separate subproblems ([Goel and Pirolli, 1992], [Pople, 1982], [Newell and Simon, 1995]). Problems in ill-defined domains however, do not necessarily decompose into subproblems that are independent or easier to solve [Simon, 1973].

The swallow problem, for example, may be solved using backward chaining ([VanLehn et al., 2005], [VanLehn et al., 2004]). Given the sought quantity, we may identify a principle from Newtonian Mechanics that defines it in terms of other quantities such as the mass of the swallow and its acceleration. Each of these quantities may be solved independently of the others. While their values are related, it is unimportant how they are identified so long as it is done accurately.

Now consider the problem of designing a house for the swallow. We could divide it into subproblems such as choosing a site, or determining the size of the house to be built. These problems, however, are not independent. Selecting a given site limits us to houses that will fit on it. Selecting a given size in turn limits the sites we may choose. The answer to one subproblem necessarily constrains the other, and neither one may be solved without considering the effect it has on the other.

Several authors ([Pople, 1982] [Goel and Pirolli, 1992], [Simon, 1973], [Reitman, 1964]) have noted that human problem solvers often cope with this by solving the subproblems in parallel. As each problem-solving step is taken, it is evaluated both in terms of the current subproblem and the constraints it imposes on the others. Unlike the independent decomposition of well-defined problems this decomposition into interrelated subproblems does not reduce the complexity of the overall problem-solving process. Nevertheless such behavior is important in domains such as writing, as illustrated here, where the paper is organized into sections with each one framing and constraining the contents of the next.

HUMAN TUTORING STRATEGIES

Students in well-defined domains are commonly trained using batteries of practice problems followed by tests which are solved by and checked against a formal theory. While there has been some effort to move away from this approach [Callahan and Hoffman, 1995] it is still common practice. In this section we briefly summarize several strategies employed by human tutors in ill-defined domains.

4.1 Case Studies

Engineering students who study applied ethics routinely study real cases faced by engineering practitioners ([Ashley et al., 2004], [Harris et al., 1995]). Law students study how legal rules have been applied in past decisions, briefs, and arguments ([Llewellyn, 1981], [Aleven et al., 2005] [Ashley et al., 2005]). Architecture students likewise examine real buildings to identify both successes and failures ([Brolin, 1976], [Mullet and Sano, 1995]). Such examples can often highlight the constraints and nuances of an ill-defined domain far better than any abstract model. They also provide the students with analogies on which to base their own work [Llewellyn, 1981]. Williams [Williams, 1992] presents a conceptual analysis of this style of learning in both the legal and medical domains.

Examples are also used for instruction in well-defined domains. However those cases are unambiguously correct or incorrect and do not require any interpretation. By contrast, in ill-defined domains, the process of interpreting case examples is often carried out by means of a Socratic dialogue where students are guided by the instructor's questions.

This process of interpretation is often carried out by means of a Socratic dialogue where students are guided by the instructor's questions.

4.2 Weak Theory Scaffolding

Architecture students at the University of Oregon are directed to read *A Pattern Language* and *The Timeless Way of Building* ([Alexander, 1977] [Alexander, 1979]). They then describe existing buildings in terms of the language and design new ones according to its precepts. They do so with the knowledge that the theory itself is incomplete. As they advance, the theory's restrictions are faded out until students need not follow them at all.

This process is advantageous in that it gives the students a conceptual framework to structure the ill-defined domains. It enables them to ignore some challenges, while tackling others. In this way the domain is made more tractable without being entirely constrained. Similar strategies have also been employed in well-defined domains such as physics and probability [VanLehn et al., 2004].

4.3 Expert Review

Students in well-defined domains such as physics may test their results against a formal theory. In ill-defined domains no absolute theory exists. Students in these domains typically submit their work to domain experts for comment. Law students routinely engage in "moot court" sessions where they present arguments in a real case before judges or law professors, who critique the sessions and provide the students with immediate feedback. Students in architecture likewise typically submit their designs to faculty panels for review.

4.4 Peer Review/Collaboration

The "Crit" is a longstanding practice in architecture education. Like "brown-bag talks" in research fields, it is a chance for students to learn from their peers. During a crit, students present finished work to their peers for comments. While their peers may not be experts in the field, they have unique perspectives on the problem and can often provide novel feedback. While such advice may not be objectively verifiable, experience has shown that it is often quite valuable.

ITS STRATEGIES

In this section we will describe some ITS techniques that have been employed in ill-defined domains. We provide representative examples of systems employing each strategy. Although some of the strategies lack any apparent "artificial intelligence", they have proven successful in past studies. As Burke and Kass astutely noted, it is possible for a relatively ignorant system to teach complex ideas:

> "It is important to keep in mind that the domain knowledge, which must be conveyed to the student, does *not* have to be the same knowledge base that the system uses to diagnose the student's misconceptions. It is possible to build teaching systems that can effectively convey domain knowledge that is much richer and more complex than the system itself can understand. It is crucial to build upon this insight if intelligent tutoring is going to move beyond the easily-formalized domains, such as arithmetic" [Burke and Kass, 1996].

5.1 Model-Based

In a model-based ITS, instruction is based upon ideal solution models. These models represent one or more acceptable solutions to a given problem, or a general model of the domain as a whole. The model is used both to check the students' actions and provide help. Model tracing tutors may be loosely classified as either *strong* or *weak*. Strong tutors force the students to follow the model exactly so that each entry matches it in some way. The Andes ITS for physics uses this methodology. Each problem is represented by a solution model that contains all correct entries. Every student entry must be represented in the model in order to be accepted. Such systems are typically called model-tracing tutors. Weak methods use the model as a guide but do not require strict adherence to its contents.

Strong model-based tutors have proven successful in well-defined domains such as physics [VanLehn et al., 2005]. They have not yet achieved the same success in ill-defined domains. Doing so would require formalizing some model of the domain such as an analogous game [Allen and Saxon, 1998] or a subdomain model ([Ashley, 2000], [Aleven, 2003], [Gardner, 1987]). Such a tutor would then be operating in a comparatively well-defined subset of the domain. As Burke and Kass suggest, it still may be possible for that tutor to teach skills in the subset in a way that leaves open key ill-defined elements so as not to mislead students.

Such 'weaker' model-based approaches have achieved some success in ill-defined domains. The CATO system ([Aleven, 2003], [Ashley, 2000]) used a model of the domain to relate real cases and argument models to present to students. PETE [Goldin et al., 2001] uses a weak domain model to teach engineering ethics. While neither of these are model-tracing systems, they do make use of formal models as guides and show that such models may be useful in ill-defined domains.

5.2 Constraints

While model-tracing systems are based upon a complete solution or domain model, constraint-based systems are built from sets of constraints. These constraints specify what characteristics a solution should, or should not have. These requirements may be sufficient to provide a complete solution specification or only a partial description. Constraints may be classified as either *strong* or *weak*. Strong constraints represent absolute requirements or prohibitions. Weak constraints represent preferences, or warnings. Unlike model-based systems, these constraint-based systems are not based upon complete, or necessarily consistent models of the domain in question.

Strong constraints have been employed in music [Holland, 1999] and [Brandao, 2005]. These systems permit students to define unique musical combinations but prevent them from violating well-accepted rules of harmony and tone. This work echoed Reitman's intuition [Reitman, 1964] that, while it is difficult to tell what *is* a fugue, it is often easy to tell what *is not*.

Weak constraints have been employed in ill-defined domains but to a lesser degree. We have begun work on a legal mark-up system that employs weak constraints to guide students [Pinkwart et al., 2006]. While students are largely free to specify the relationships they wish, the system will coach them on "optimal" choices using a set of solution preferences. Suthers [Suthers, 1998] used a similar approach but based his constraints upon expert solutions.

Simon [Simon, 1973], Reitman [Reitman, 1964], and Goel [Goel and Pirolli, 1992] all speak of problem solvers propagating constraints as they work. This process allows them to bound and narrow the search space to manageable dimensions. This view mirrors Alexander's comments in *A Pattern Language* and *The Timeless Way of Building* ([Alexander, 1977] [Alexander, 1979]) that individual patterns such as appropriate housing locations necessarily constrain the design of housing. The Pattern Language in effect introduces a set of constraints that a practitioner may choose to follow. As the practitioner makes decisions those decisions "activate" an individual pattern and propagate its constraints for future decisions. This similarity may be an indication that constraint-based tutoring is pedagogically more appropriate for ill-defined domains than model-tracing methods. The question remains whether a constraint-based tutor can provide all of the feedback that is appropriate.

5.3 Discovery Learning

Instruction in many domains consists of reifying or formalizing domain knowledge and then transmitting it explicitly via texts or lectures. Researchers like Seymour Papert have long argued for a more constructivist approach. This approach is variously known as LOGO ([Resnick, 1998], [Holland, 1999]), Discovery Learning ([Veermans et al., 2006], [van Joolingen, 1999], [de Jong and van Joolingen, 1998] [Veermans and van Joolingen, 2004]), or Discovery Microworlds [Trafton and Trickett, 2001]. For the purposes of this paper we will group these under the heading of "Discovery Learning." De Jong [de Jong and van Joolingen, 1998] and van Joolingen [van Joolingen, 1999] have provided some useful comparative analyses of discovery learning. Their comparisons indicate that, although the method can be successful, students may have difficulties with the methodology nullifying their learning gains.

Discovery Learning approaches can be classified into three general tracks: discovery support; model exploration; and model building. *Discovery Support* systems operate by providing the user with support as they work on a task in an unconstrained domain. Such systems do not attempt to model the domain itself. Rather they seek to assist the students in exploring the domain by providing intelligent support. The HYPO system provides users with intelligent case-retrieval tools to help in issue spotting [Ashley, 1990]. Other systems have focused on providing intelligent suggestions for data mining tasks [Bernstein et al., 2005] and hypothesis support for basic science education [Suthers, 1998]. In general, any decision support system could qualify as a discovery support model so long as it helped the users better understand the domain and did not supplant their own actions.

Model Exploration systems also focus on helping students to explore a domain. Unlike discovery support systems the domain in question is represented by a formal model. Here students interact with a model of the domain rather than conducting real-world searches or experiments, drawing conclusions from it that are appropriate to the real domain. This technique has been used in some well-defined subdomains of physics ([Veermans et al., 2006], [Veermans and van Joolingen, 2004], [Trafton and Trickett, 2001]). Apart from some work in music [Holland, 1999], it has yet to be applied to any ill-defined domains, due in large part to the challenges of modeling such domains.

Model Building by contrast focuses on the development of domain models. Model building systems supply users with a suite of model building tools such as a formal development language and (optionally) some support in their use. The most prevalent such tools are descendants of Papert's original LOGO programming language such as StarLogo [Resnick, 1998] and Music Logo [Holland, 1999]. Recent work by Yoshino and Sakurai has extended this principle to law via a logic-programming language [Yoshino and Sakurai, 2005]. Model building has a long history in psychological research including the study of problem-solving in ill-defined domains [Voss and Post, 1988].

By providing an open-ended basis for experimentation, these systems enable users to test their own intuitions about a domain and to perform arbitrary experiments. In so doing it is believed that students will better understand nuances of the domain and the challenges of modeling it that a premade system would hide. LOGO-like languages are also easier to develop than full-scale domain simulations.

Of these three methodologies, Model-Building and Discovery Support seem to hold the most promise for ill-defined domains. Both follow Burke and Kass' intuition above and permit the user to explore the nuances of a domain without limiting them or hiding the relevant aspects.

5.4 Case Analysis

As stated above, the examination of past cases is a primary teaching tool in ill-defined domains such as law, architecture and music. Systems that facilitate this process by providing cases, highlighting their relationships or otherwise facilitating analysis have long been a part of ITS research.

Ashley's HYPO [Ashley, 1990] and it's descendant CATO [Aleven and Ashley, 1996] were built on a relatively formal, though nondeterministic, model of trade secret law. This model was used to encode real cases in terms of their relevant legal factors. A hierarchy of factors was then used to categorize cases and present them to the students as they developed an argument structure. The CATO-Dial system [Ashley et al., 2002] built upon this to engage the students in court-like arguments.

Burke and Kass [Burke and Kass, 1996] also developed an automatic case-retrieval system to select cases in order to present feedback based upon the most recent student action. While their domain model was quite limited, the use of real cases enabled them to give more nuanced feedback than the system could otherwise describe.

Relatively simple techniques such as self-explanation prompts ([Schworm and Renkl, 2002], [Aleven et al., 2005], [Pinkwart et al., 2006]) have also been shown to boost student performance on case-analysis tasks. We believe that these techniques may be fruitfully combined with case retrieval systems such as those above to improve learning gains.

5.5 Collaboration

Education researchers such as Vygotsky have long argued that social support for learning is as or more important than instructional factors ([Chan and Baskin, 1990] [Chan et al., 1993]). Engaging students with similarly-situated peers can encourage their own learning and that of their peers. Using Crit-like techniques in design tasks, can give students the benefit of alternate, sometimes intensely critical, perspectives. Existing collaborative ITSs have provided this support either by casting the system as collaborator, or by using the system to facilitate interactions among human peers.

The former method has been championed by Tak-Wai Chan ([Chan and Baskin, 1990] [Chan et al., 1993]) and his colleagues. In this approach the system is equipped with a user model and cast as a student alongside the real student. The virtual student is designed to be as mistake-prone as its human peers and to learn alongside them. The challenge lies in developing a rational student model for the virtual student. While this may be easier than an expert model, it is just as critical. An irrational or unbelievable student model may, over time, cancel out any purported benefits. Although this approach has yet to be applied to ill-defined domains, it seems to us that the approach is applicable in that it resembles Socratic Dialogue.

The second approach, while relatively AI-free, has gained acceptance in recent years. The role of the system is to facilitate user interaction and leverage the users' collective knowledge for learning. This approach follows Burke and Kass' intuition that a relatively ignorant system can nevertheless support learning gains. Soller et al. [Soller et al., 2005] provides a nice summary of the state of the art in this area. In recent years systems have been developed along these lines for writing [Cho and Schunn, 2005], collaborative discovery [Suthers, 1998], linguistics [Weisler et al., 2001], and mediation [Tanaka et al., 2005]. Our own group has begun working on just such a system for legal argument formation and analysis [Pinkwart et al., 2006].

CONCLUSION

Given the distinct characteristics, teaching techniques, and ITS strategies, discussed above, it is our opinion that there are substantive differences between well-defined and ill-defined domains. Those differences should be taken into account by researchers in this area. Having said that, we believe that the future is promising. Research in this arena has already begun to establish a solid (though ill-defined) foundation for future work. Given the nature of this arena it seems inappropriate to draw any "final" conclusions at this time. Rather we will close with two key challenges.

Firstly, the community should continue to develop new tutoring strategies and seek out ways to combine existing strategies. In our opinion, hybrid systems hold the most promise for ITSs in ill-defined domains, especially those that build upon Burke and Kass' intuitions.

Secondly, and most importantly, we urge the community to consider focusing across domains. Ill-defined domains share many common characteristics such as open-textured concepts and a lack of absolute verification. Each individual domain such as law or architecture may possess these characteristics to a different degree. Moreover, such domains are themselves quite amorphous. They contain many distinct "tasks" or subdomains, such as the formation of legal arguments, and the deciding of legal cases, that each have their own characteristics and requirements.

In our opinion, asking what ITS strategies are appropriate for "the law" is too narrow a question. Rather, researchers should focus on tutoring strategies that are appropriate for general characteristics such as overlapping subproblems and open-textured concepts. It is our belief that framing research questions in this way will lend itself to the development of strategies that can be "ported" across ill-defined and even well-defined domains. Of course, insights can be gained by looking at particular domains and tasks, but the true significance of these insights can best be appreciated by looking at characteristics that cross domains.

We do not expect this to be the last word on the subject. There are other challenges facing the ill-defined ITS community. Rather we hope that this survey will guide strategies that can spur research in new directions.

REFERENCES

[Aleven, 2003] Aleven, V. (2003).Using background knowledge in case-based legal reasoning: A computational model and an intelligent learning environment.*AI*, 150:183–237.

[Aleven and Ashley, 1996] Aleven, V. and Ashley, K. D. (1996).How different is different? arguing about the significance of similarities and differences.In Smith, I. and Faltings, B., editors, *Advances in CBR: Proc. of the 3rd European Workshop, EWCBR-96, LNAI*, pages 1–15. Springer Verlag: Berlin.

[Aleven et al., 2005] Aleven, V., Ashley, K. D., and Lynch, C. (2005).Helping law students to understand supreme court oral arguments: A planned experiment.In Sartor, G., editor, *ICAIL*, pages 55–59. ACM, New York.

[Alexander, 1977] Alexander, C. (1977).*A Pattern Language*.Oxford University Press; US.

[Alexander, 1979] Alexander, C. (1979).*The Timeless Way of Building*.Oxford University Press; US.

[Allen and Saxon, 1998] Allen, L. and Saxon, C. S. (1998).The legal argument game of legal relations.*E-Law - Murdoch U. Electronic Journal of Law*, 5(3).

[Ashley, 1990] Ashley, K. D. (1990).*Modeling Legal Argument: Reasoning With Cases and Hypotheticals.*The Bradford Books, MIT Press.

[Ashley, 2000] Ashley, K. D. (2000).Designing electronic casebooks that talk back: The cato program.*Jurimetrics*, 40(3):275–319.

[Ashley et al., 2005] Ashley, K. D., Aleven, V., and Lynch, C. (2005).Teaching creative legal reasoning with examples from supreme court oral arguments.In Yoshino, H., Ashley, K. D., and Nitta, K., editors, *AILE; Proc. of the ICAIL-05 Workshop.*

[Ashley et al., 2004] Ashley, K. D., Chi, M., Pinkus, R., and Moore, J. D. (2004).Modeling learning to reason with cases in engineering ethics: A test domain for intelligent assistance. NSF Proposal.

[Ashley et al., 2002] Ashley, K. D., Desai, R., and Levine, J. M. (2002).Teaching case-based argumentation concepts using dialectic arguments vs. didactic explanations.In Cerri, S. A., Gouardères, G., and Paraguaçu, F., editors, *ITS*, volume 2363 of *LNCS*, pages 585–595. Springer.

[Berman and Hafner, 1985] Berman, D. and Hafner, C. (1985).Obstacles to the development of logic-based models of legal reasoning.In Walter, C. and Allen, L. E., editors, *Computing Power & Legal Reasoning*. West Pub Co. St Paul.

[Bernstein et al., 2005] Bernstein, A., Hill, S., and Provost, F. (2005).Intelligent assistance for the data mining process; an ontology-based approach.*IEEE Transactions on Knowledge and Data Engineering*, 17(4):503–518.

[Brandao, 2005] Brandao, M. (2005).Guided discovery tutoring and schoenberg's harmony teaching method: an investigation.

[Brolin, 1976] Brolin, B. C. (1976).*The Failure of Modern Architecture*.Van Nostrand Reinhold Company.

[Burke and Kass, 1996] Burke, R. D. and Kass, A. (1996).Interest-focused tutoring: A tractable approach to modeling in intelligent tutoring systems.Technical Report TR-96-08.

[Callahan and Hoffman, 1995] Callahan, J. and Hoffman, K. (1995).*Calculus In Context*.W.H. Freeman and Co. New York.

[Chan and Baskin, 1990] Chan, T. W. and Baskin, A. B. (1990).Learning companion systems.In Frasson, C. and Gauthier, G., editors, *ITS*. Ablex, New Jersey.

[Chan et al., 1993] Chan, T. W., Lin, C. C., Lin, S. J., and Kuo, H. C. (1993).Octr: A model of learning stages.In *Proc. of AI-Ed, Edinburgh U.K.*, pages 257–264.

[Cho and Schunn, 2005] Cho, K. and Schunn, C. D. (2005).Scaffolded writing and rewriting in the discipline: A web-based reciprocal peer review system.*Computers & Education (In Press)*.

[de Jong and van Joolingen, 1998] de Jong, T. and van Joolingen, W. (1998).Scientific discovery learning with computer simulations of conceptual domains.*Review of Educational Research*, 68(2):179–201.

[Gardner, 1987] Gardner, A. V. D. L. (1987).*An Artificial Intelligence Approach To Legal Reasoning.*The Bradford Books, MIT Press.

[Goel and Pirolli, 1992] Goel, V. and Pirolli, P. (1992).The structure of design problem spaces.*Cognitive Sci.*, 16:345–429.

[Goldin et al., 2001] Goldin, I. M., Ashley, K. D., and Pinkus, R. L. (2001).Introducing PETE: computer support for teaching ethics.In *ICAIL*, pages 94–98.

[Harris et al., 1995] Harris, C. E., Pritchard, M. S., and Rabins, M. J. (1995).*Engineering Ethics*.Wadsworth Publishing Co, Boston.

[Holland, 1999] Holland, S. (1999).Artificial intelligence in music education: A critical review.*Readings in Music and AI; Contemporary Music Studies*, 20.

[Johnson, 1988] Johnson, E. J. (1988).*Expertise and Decision Under Uncertainty: Performance and Process*, chapter 7.Lawerence Erlbaum Associates: Hillsdale New Jersey.

[Lawrence, 1988] Lawrence, J. A. (1988).*Expertise on the Bench: Modeling Magistrates' Judicial Decisionmaking*, chapter 8.Lawerence Erlbaum Associates: Hillsdale New Jersey.

[Lehmann et al., 2005] Lehmann, J., Biasiotti, M. A., Francesconi, E., and Sagri, M. T., editors (2005).*LOAIT: Proc. of the ICAIL-05 Workshop*, IAAIL Workshop Series. Wolf Legal Publishers.

[Lester et al., 2004] Lester, J. C., Vicari, R. M., and Paraguaçu, F., editors (2004).*Intelligent Tutoring Systems, 7th International Conference, ITS 2004, Maceiò, Alagoas, Brazil, August 30 - September 3, 2004, Proceedings*, volume 3220 of *LNCS*. Springer.

[Levi, 1949] Levi, E. H. (1949).*An Introduction to Legal Reasoning.*U. Chicago Press; Chicago.

[Llewellyn, 1981] Llewellyn, K. N. (1981).*The Bramble Bush; On Our Law and its Study.*Oceana Publications Inc. Dobbs Ferry, New York.

[Minsky, 1995] Minsky, M. (1995).Steps to artificial intelligence.In Luger, G. F., editor, *Computation and Intelligence, Collected Readings.*, chapter 3, pages 47–90. AAAI, Menlo Park CA, and MIT Press.

[Mullet and Sano, 1995] Mullet, K. and Sano, D. (1995).*Designing Visual Interfaces*.Sun Microsystems Inc. Mountain View California.

[Newell, 1969] Newell, A. (1969).Heuristic programming: Ill-structured problems.*Progress in Operations Research*, 3:361–413.

[Newell and Simon, 1995] Newell, A. and Simon, H. A. (1995).Gps, a program that simulates human thought.In Luger, G. F., editor, *Computation and Intelligence, Collected Readings.*, chapter 16, pages 415–428. AAAI, and MIT Press.

[Pinkwart et al., 2006] Pinkwart, N., Aleven, V., Ashley, K. D., and Lynch, C. (2006).Toward legal argument instruction with graph grammars and collaborative filtering techniques.In Ashley, K. D. and Ikeda, M., editors, *ITS06, Proc. of the 2006 Conference on ITS (In-Press)*.

[Pople, 1982] Pople, H. E. (1982).*Heuristic Methods for Imposing Structure on Ill-Structured Problems: The Structuring of Medical Diagnostics*, chapter 5.Westview Press: Boulder Colorado.

[Radin, 1933] Radin, M. (1933).Case law and stare decisis concerning präjudizienrecht in amerika.*Columbia Law Review*, 27(2):199–212.

[Reitman, 1964] Reitman, W. R. (1964).*Heuristic Decision Procedures Open Constraints and the Structure of Ill-Defined Problems*, chapter 15, pages 282–315.John Wiley & Sons Inc. New York.

[Resnick, 1998] Resnick, M. (1998).*Learning About Life*, chapter 11, pages 229–241.MIT Press.

[Schauer, 1998] Schauer, F. (1998).Prediction and particularity.*Boston University Law Review*.

[Schworm and Renkl, 2002] Schworm, S. and Renkl, A. (2002).Learning by solved example problems: Instructional explanations reduce self explanation activity.In *Proc. of the 24th Annual Conf. of the Cognitive Science Society, Mahwah NJ*, pages 816–821. Lawrence Erlbaum.

[Sergot et al., 1986] Sergot, M. J., Sadri, F., Kowalski, R. A., Kriwaczek, F., Hammond, P., and Cory, H. T. (1986).The british nationality act as a logic program.*Comm. ACM*, 29(5):370–386.

[Simon, 1973] Simon, H. A. (1973).The structure of ill-structured problems.*AI*, 4:181–201.

[Soller et al., 2005] Soller, A., Martinez, A., Jermann, P., and Muehlenbrock, M. (2005).From mirroring to guiding: A review of state of the art technology for supporting collaborative learning.*International Journal of AI-Ed*, 15:261–290.

[Suthers, 1998] Suthers, D. (1998).Representations for scaffolding collaborative inquiry on ill-structured problems.

[Tanaka et al., 2005] Tanaka, T., Yasumura, Y., Kaagami, D., and Nitta, K. (2005).Case based online training support system for adr mediator.In Yoshino, H., Ashley, K. D., and Nitta, K., editors, *AILE; Proc. of the ICAIL-05 Workshop*.

[Trafton and Trickett, 2001] Trafton, J. G. and Trickett, S. B. (2001).Note-taking for self-explanation and problem solving.*Human-Computer Interaction*, 16:1–38.

[van Joolingen, 1999] van Joolingen, W. (1999).Cognitive tools for discovery learning.*International Journal of AIEd*, 10:385–397.

[VanLehn et al., 2004] VanLehn, K., Bhembe, D., Chi, M., Lynch, C., Schulze, K. G., Shelby, R., Taylor, L., Treacy, D., Weinstein, A., and Wintersgill, M. (2004).Implicit versus explicit learning of strategies in a non-procedural cognitive skill.In [Lester et al., 2004], pages 521–530.

[VanLehn et al., 2005] VanLehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R., Taylor, L., Treacy, D., Weinstein, A., and Wintersgill, M. (2005).The andes physics tutoring system: Lessons learned.*International Journal of AI-Ed*, 15(3).

[Veermans et al., 2006] Veermans, K., van Joolingen, W., and de Jong, T. (2006).Use of heuristics to facilitate scientific discovery learning in a simulation learning environment in a physics domain.*International Journal of Science Ed.*, 28(4):341–361.

[Veermans and van Joolingen, 2004] Veermans, K. and van Joolingen, W. R. (2004).Combining heuristics and formal methods in a tool for supporting simulation-based discovery learning.In [Lester et al., 2004], pages 217–226.

[Voss and Post, 1988] Voss, J. F. and Post, T. A. (1988).*On the Solving of Ill-Structured Problems*, chapter 9.Lawerence Erlbaum Associates: Hillsdale New Jersey.

[Weisler et al., 2001] Weisler, S., Bellin, R., Spector, L., and Stillings, N. (2001).An inquiry-based approach to e-learning: The chat digital learning environment.In *Proceedings of SSGRR-2001. Scuola Superiore G. Reiss Romoli, L'Aquila, Italy.*

[Williams, 1992] Williams, S. M. (1992).Putting case-based instruction into context: Examples from legal and medical education.*The Journal of the Learning Sciences*, 2(4):367–427.

[Yoshino and Sakurai, 2005] Yoshino, H. and Sakurai, S. (2005).A knowledge-based systems approach to educating creative legal minds.In Yoshino, H., Ashley, K. D., and Nitta, K., editors, *AILE; Proc. of the ICAIL-05 Workshop.*