Guest Editor's Introduction

KBS Validation: From Tools to Methodology

Enric Plaza I Cervera, Institut d'Investigació en Intel·ligència Artificial

THE FIELD OF KNOWLEDGE-BASED SYSTEMS HAS LACKED A COMPREHENSIVE VIEW OF KBS VALIDATION AND THE ROLE OF V&V IN THE KBS LIFE CYCLE. BY PRESENTING KEY WORK FROM EUROVAV '91, THIS SPECIAL SERIES HIGHLIGHTS AND EXAMINES SIGNIFICANT TRENDS IN THIS IMPORTANT AREA.

KNOWLEDGE-BASED SYSTEMS have outgrown academic applications, and their widespread use in industry has boosted work on the methodology required to design, implement, and validate them. Although knowledge acquisition was a prominent focus of research and practice from the beginning of this industrial push, only recently have the issues of the KBS life cycle and verification and validation been put at the same level of discussion. During the first years of KBS production, researchers thought of the KBS life cycle as rapid prototyping, while V&V was handled on an as-needed basis. Soon methods (and their support tools) were developed to detect problems in rule-based systems, such as redundant, subsumed, or missing rules. However, the field lacked a comprehensive view of KBS validation and the role of V&V in the KBS life cycle.[1] European developers sensed this need and decided to address it at a workshop at the 1990 European Conference on AI, and through ESPRIT's various KBS projects.

It was in this context that the European Workshop on Verification and Validation of Knowledge-Based Systems (EuroVAV '91) took place at Jesus College in Cambridge, England, in July 1991. About half of the 58 delegates were from industry, and half from academia. Presentations were given in three tracks: theoretical foundations for V&V, tools and techniques, and industrial requirements. There were also lively daily discussion groups delving into the issues that had arisen during the regular sessions. By presenting key work from EuroVAV '91, *IEEE Expert*'s special series on V&V for KBSs highlights and examines significant trends in this important area.

A recurrent discussion throughout EuroVAV '91, as in other V&V workshops, was on terminology. V&V researchers and developers often use the same terms but with different meanings, so it is easy to misunderstand each other. This is to be expected in a field that lacks an established set of concepts and terms. In their article on pp. 48-55, Thomas Hoppe and Pedro Meseguer explain some of the difficulties as well as the approaches suggested, and propose a common terminology. The issues underlying their work are not merely definitional disagreements, but essential questions; for example, what does it mean in general for a KBS to be valid for a specific task?

Several other theoretical questions arose during the conference but remained unanswered. For instance, can an AI system be valid with respect to its knowledge, in the sense that it behaves as rationally as can be expected given its limited knowledge base, while not being valid with respect to performing adequately in a task environment? And if future AI systems can modify themselves to the task environment through learning or adaptation, how can their validity be defined?

0885/9000/93/0600-0045 $3.00 © 1993 IEEE

One topic strongly debated the first day was the relation of KBS V&V to the software engineering methodology for program correctness and evaluation. The discussion dealt with the usability and usefulness of formal program specifications for expert systems. Some participants said that KBSs are dedicated to complex or ill-defined problems, whose nature escapes formal specification and so suggests the use of AI techniques instead of software engineering techniques. Others argued for the partial usefulness of formal specification in KBS design, since verification always occurs with respect to some specification. They argued for adapting software engineering techniques to the KBS field. The two camps finally agreed that some specification was needed, and that two avenues for research were open. The first involves using and adapting the formal-specification approach: The article by Paul Krause, John Fox, Mike O'Neil, and Andrzej Glowinski on pp. 56-61 portrays practical experience using formal specification for medical decision support systems.

A second approach is the use of conceptual modeling frameworks such as KADS[2] and the Components of Expertise[3] as a new kind of specification for KBS design. Luc Steels presented the Components of Expertise framework in the workshop's keynote speech, and argued for the knowledge-level models of KBS based on abstract entities such as tasks, methods, and models. Knowledge models provide a framework in which to include the V&V of pragmatic constraints, and the progressive refinement of knowledge models provides a principled mapping to the symbol level necessary for V&V. This knowledge-level approach to KBS construction is favored in current European projects funded by ESPRIT. For instance, the article by Ole Mengshoel on pp. 62-68 describes the KVAT knowledge validation tool, used during knowledge acquisition in the context of the KADS methodology. To check on the validity of knowledge base contents, KVAT uses specific cases that summarize the problem-solving functionality required, thus focusing further knowledge acquisition.

Another debate group discussed the relation of V&V to existing KBS development methodologies. The group agreed there is a strong need to harmonize these methodologies with practical V&V techniques and tools, and that the main lesson to be learned from software engineering was the systematic and controlled use of V&V methods and tools throughout the KBS life cycle.

The second day of the workshop was dedicated to tools and techniques. Although progress has been made toward practical tools, some explicitly intended to be generic, we are still some distance from having a standard set of tools and techniques covering the complete KBS life cycle. The tools and techniques presented at the workshop define different ranges of applicability and check different sets of properties. For instance, the UVT tool described in the article by Faruk Polat and H. Altay Guvenir on pp. 69-75 checks several properties of knowledge bases that use unification as the inference procedure. This technique could also be applied to other KBSs that conform to the assumptions made by UVT about their knowledge bases and the use of unification. Granting those assumptions, UVT is generic in the sense that is applicable to that set of KBSs but it is specialized in checking some properties (rule redundancy, conflict, subsumption, circularity, and so on) and not others (such as semantic inconsistency among clauses). One of the main conclusions on the second day was that there is no overall methodology for selecting and organizing V&V tools and techniques. The participants saw this as a big drawback in the state of the art, and an important one that requires further attention.

WE NEED FRAMEWORKS FOR VERIFICATION, VALIDATION, AND EVALUATION TECHNIQUES AND TOOLS THAT ARE NOT MERELY TOOLKITS.

Another interesting and active topic of discussion was knowledge acquisition for validation. Each technique presented at the workshop requires that some domain knowledge be acquired specifically for validation. The most common example is a simple set of test cases, but even that requires a deep understanding of the domain to assure the selection of test sets, the definition of correct or "acceptable" solutions, and so on. Other tools require the acquisition of different sorts of semantic constraints from the domain. The discussion group concluded that knowledge acquisition has to account for V&V, and it can do so by including activities that also acquire the knowledge needed in validation.

Two more discussion groups dealt with system testing and the requirements for a V&V toolkit. The first group agreed that KBS testing would likely require more time and effort to cover competency and confidence than conventional software systems. Several approaches were listed, and it was stressed that testing should not be a separate and final stage, but used in combination with structural-validation techniques such as static and dynamic inspection. The desiderata for a V&V toolkit make up a long list, but people agreed that its design should go hand in hand with the development of life cycle protocols and KBS development methodologies such as the knowledge level or conceptual frameworks. An example of this trend is the work of Trevor Bench-Capon, Frans Coenen, Hyacinth Nwana, Ray Paton, and Michael Shave, described in the article on pp. 76-81. Their team developed the Mekas methodology, which helps KBS developers acquire, maintain, and revise knowledge bases, and then developed the MAKE environment in that framework.

The third day's sessions on industrial requirements confirmed that we are at an early stage in the use of V&V tools and techniques in real applications. The article by Paul Krause and colleagues (mentioned earlier) investigates whether medical decision support systems can be specified formally using the Z language. The authors conclude that formal representation is not yet possible because of the state of medical knowledge in many domains, but partial formalization is feasible. More practical approaches are possible for medial KBSs: Pedro Meseguer has elsewhere presented a method based on an extension of assumption-based truth maintenance labels that has successfully verified several kinds of properties in a medical expert system for pneumonia diagnosis (this work won the 1992 ECAI prize).[4]

Discussion groups about industrial requirements tackled three issues. The first was how to specify V&V requirements and define acceptance criteria for KBSs. All agreed that this is difficult but necessary.

For KBSs, the nature of expertise is intrinsically linked to a specification, and there must be mechanisms for representing and developing this relationship within a V&V framework. This should occur when the project is first defined and then evolve in a controlled way. Another discussion group analyzed V&V's relationship with systems evaluation, concluding that the former is a component of the latter. This group also recognized that models of users and tasks are extremely important. The third group dealt with V&V in relation to quality assurance, safety-critical standards, and system certification. They decided that V&V fits increasingly into these areas but that much work remains before deploying critical KBS applications and developing standards.

As the "future-note" speaker, I closed the EuroVAV '91 workshop by exploring several V&V trends based on the workshop presentations and discussion groups:

- We need frameworks for verification, validation, and evaluation techniques and tools that are not merely toolkits.
- Formal specifications are useful for knowledge engineering and KBS development, although their scope and applicability must be clarified through practice.
- Conceptual-modeling frameworks and knowledge-level models of KBS tasks and expert systems comprise a growing trend in KBS development, and can be used to develop more high-level validation techniques that are closer to knowledge validation than more syntactic-based V&V tools.
- KBS life cycle protocols are needed to establish which and when validation activities are to be performed in consonance with overall KBS development methodologies.
- Current research on standardization[5] might be relevant to V&V: Generic tasks and methods can have associated validation methods as well as associated knowledge acquisition tools. Shared domain ontologies and interlingua research might prove as important to V&V as they are to knowledge acquisition.
- We must improve representation languages so that they can support incremental verification and validation. This improvement might include verification by construction, as in software engineering, as well as local, modular verification methods.

I hope the articles in this special series help clarify many of these issues, and point to further needed work.

Acknowledgments

This research was part of the Valid Project, performed at the Spanish Council for Scientific Research's Institut d'Investigació en Intel·ligència Artificial and funded by the ESPRIT-II program of the Commission of the European Communities, project number 2148.

References

1. B. Lopez, P. Meseguer, and E. Plaza, "Knowledge-Based Systems Validation: A State of the Art," *Artificial Intelligence Communications*, Vol. 3, No. 2, June 1990, pp. 56-72.
2. B. Wielinga, A. Schreiber, and J. Breuker, "KADS: A Modelling Approach to Knowledge Engineering," *Knowledge Acquisition*, Vol. 4, No. 1, 1992. .
3. L. Steels, "The Components of Expertise," *AI Magazine*, Vol. 11, No. 2, 1991, pp. 28-49.
4. P. Meseguer, "Incremental Verification of Rule-Based Expert Systems," *Proc. 10th European Conf. on Artificial Intelligence (ECAI '92)*, John Wiley & Sons, New York, 1992, pp. 840-844.
5. R. Neches et al., "Enabling Technology for Knowledge Sharing," *AI Magazine*, Vol. 12, No. 3, 1991, pp. 37-56.

Enric Plaza i Cervera is a research scientist at the Institut d'Investigació en Intel·ligència Artificial, an institute of the Spanish Council for Scientific Research. His research focuses on knowledge acquisition for KBSs, case-based reasoning and learning, KBS validation, and integrated learning architectures. His current work involves systems that integrate problem solving and learning, and representing languages with reflection capabilities as a tool for this integration. With his collaborators at IIIA, he is developing a memory-based approach to integrating inference methods such as analogy and inheritance with different learning methods. Plaza received his PhD in computer science from the Technical University of Catalonia in 1987.

The author can be reached at IIIA, CSIC, Camí de Santa Bàrbara, 17300 Blanes, Catalunya, Spain; Internet, plaza@ceab.es