*Knauf, Rainer, Jantke, Klaus P.; Gonzalez, Avelino J.; Philippow, Ilka :*

***Fundamental Considerations of Competence Assessment for Validation***

*Zuerst erschienen in:*
Proceedings of the Eleventh International Florida Artificial Intelligence Research Symposium Conference : Sanibel Island, Florida, 18 - 20 May 1998 (FLAIRS 1998) / ed. by Diane Cook. - Menlo Park, Calif. : AAAI Press, 1998, S. 457-461
*Referenz-Link AAAI:*
http://www.aaai.org/Library/Conferences/FLAIRS/FLAIRS-1998/Abstracts/flairs98-088.html

# Fundamental Considerations of Competence Assessment for Validation*

**Rainer Knauf**
Technical University of Ilmenau
Faculty of Computer Science and Automation
PO Box 10 05 65, 98684 Ilmenau, Germany
Rainer.Knauf@TheoInf.TU-Ilmenau.de

**Klaus P. Jantke**
Meme Media Laboratory
Hokkaido University
Kita-13, Nishi-8, Kita-ku, Saporro 060, Japan
jantke@meme.hokudai.ac.jp

**Avelino J. Gonzalez**
Dept. of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2450, USA
ajg@ece.engr.ucf.edu

**Ilka Philippow**
Technical University of Ilmenau
Faculty of Computer Science and Automation
PO Box 10 05 65, 98684 Ilmenau, Germany
Ilka.Philippow@TheoInf.TU-Ilmenau.dc

### Abstract

This paper deals with the very fundamentals of a so-called TURING test methodology for expert system validation which was proposed by the first author (cf. (Knauf, Gonzalez, and Philippow 1997), e.g.).

First, we survey several concepts of verification and validation. Our favoured concepts are lucidly characterized by the words that verification guarantees *to build the system right* whereas validation deals with *building the right system.*

Next, we critically inspect the thought-experiment called the TURING test. It turns out that, while this approach may not be sufficient to reveal a system's intelligence, it provides a suitable methodological background to certify a system's validity. The principles of our validation approach are surveyed.

## Introduction

This paper deals with the very fundamentals of a research program which aims at the development and application of a methodology to validate intelligent systems.

We propose a TURING test - like methodology that uses test cases to estimate AI systems' validities through systematic interrogation. Our actual and further work is focussing on

1. the systematic development of test sets,
2. the development of a methodology for system validation by systematically testing,
3. appropriate ways to express validity and to use a validity statement for system refinement.

## Validation and Verification

Here, we briefly describe some basic approaches of system validation and verification including our favoured one which is used throughout the present paper as well as in our other related publications.

O'Keefe and O'Leary (cf. (O' Keefe and O'Leary 1993)) found a quite intuitive and systematic approach to characterize and distinguish *verification* and *validation* by the two circumscriptions of *building the system right* and *building the right system*, respectively.

The first property relates a system to some specification, which provides a firm basis for the question of whether or not the system on hand is *right*. In contrast, the latter formulation asks whether or not some system is considered *right*; what somehow lies in the eye of the beholder. The essential difference is illustrated in figure 1.

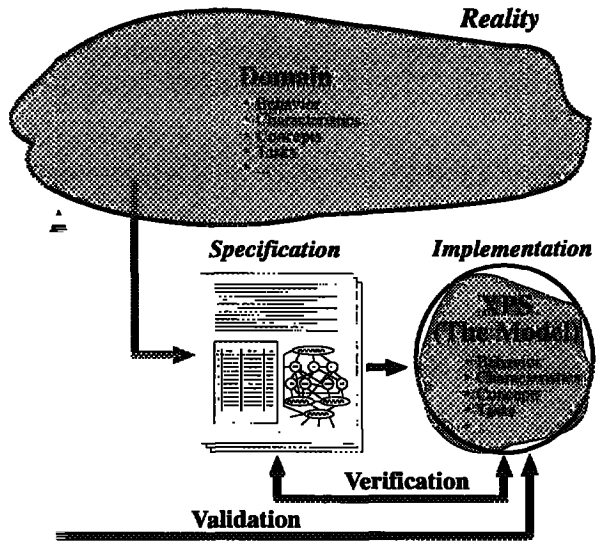


Figure 1: Validation and Verification

Throughout this paper, we adopt this perspective and put the emphasis of our investigation on the validation issue.

For AI systems, in particular, there is often no better way of evaluation than validation. Their application fields are usually characterised by not having an accepted domain model as the basis for a correct specification.

## The TURING Test as a Methaphor

The classical TURING test (cf. (Turing 1950)) means a scenario in which an artifact (a computer system) is checked for "intelligent behavior". A human interrogator is put in a dialogue situation with some remote partner not knowing whether this partner is another human or a computer system. In case the human interrogator is convinced that he is in contact with another human, this is understood to be sufficient for calling the system "intelligent". The discussion about the nature of intelligence is put off for the moment.

### Discrediting the TURING Test

Faced with the absence of sufficient consensus on the nature of intelligence, this concept certifying intelligence is specified implicitly rather than explicitly. But a closer look reveals that the question of the nature of intelligence is not just put off, it is completely missed.

Halpern's criticism (cf. (Halpern 1987)) can be summarized as follows: The so-called TURING test does not reveal anything about the inner "intelligence" of an interrogated system. It merely answers the question of whether or not a computer can be programmed to make a human conversation partner think that he is talking to another human being.

Obviously, TURING did not make an attempt towards the quantification of intelligence. The TURING test does not reveal a system's intelligence and does not seems to be an ungraded approach like this to deal with phenomena as complex as natural respectively machine intelligence.

### Analyzing the Limits of the Approach

It seems that the question of whether or not this thought-experiment may be reasonably interpreted largely depends on certain characteristics of the interrogated computer program.

For illustration, imagine an arbitrary conventional computer program that solves routine tasks with an enormous precision and within a remarkably short time. Numerical computations in arithmetics provide the simplest and quite typical instances of this application domain. Two insights are essential:

- First, these are cases in which a computer program is usually not considered to be "intelligent", although its computational power by far exceeds the intellectual power of every human being.
- Second, in these cases human interrogators will normally quite easily identify the hidden partner to be a computer and not another human being.

To sum up, there is a class of computer programs to which the TURING test approach does not apply.

More explicitly, computer programs intended to perform straightforward deterministic computations were not called intelligent, if they would behave like human experts. Even worse, such a behavior would be usually a strong indication of their incorrectness.[1]

In application domains where there exists a restricted and well-defined target behavior to be implemented, nature rarely provides better solutions. Beyond those deterministic problem domains, AI research and engineering aims at solving problems by means of computer systems that are not deterministic and less well-specified or, even worse, possibly unspecifiable. In many cases, humans deal with these problems quite adequately. Moreover, some humans are even called experts. There is not much hope to find a formal characterization of problem domains to which AI approaches typically apply.

However, we describe a certain type of problems in a partially formal way. Generally, there are several acceptable ways to react to certain input data. Given symptoms usually have multiple interpretations; in most positions on a chess board, there are several reasonable moves, and so on. Thus, such a system's correct behavior is more suitably considered a relation than a function.

Humans rarely behave functionally. Thus, programs implementing functions are inappropriate subjects of the TURING test, whereas the relational case might be revisited.

## The TURING Test Revisited - Towards a Validation Methodology

Throughout the sequel, we assume that some desired target behavior may be understood as a relation $\mathcal{R}$. With respect to some domain of admissible input data $I$ and some space of potential outputs $O$, the system's behavior is constrained by $\mathcal{R} \subseteq I \times O$ and by further problem specific conditions which might be difficult to express formally.

The validity of a system means some fitness with respect to $\mathcal{R}$. The crucial difficulty is that in most interesting application domains, the target behavior $\mathcal{R}$, in contrast to a given specification underlying verification, is not directly accessible. It might even change over time. Consequently, the illustration of figure 1, which was intended to relate and to distinguish validation and verification, needs some refinement.

At a first glance, it seems that the concept of validity refers to building the right system with respect to $\mathcal{R}$. This might be the ultimate goal, indeed.

However, the real situation illustrated by figure 2 suggests that there are intermediary between reality and the AI system under investigation, the experts. Consequently, one has to measure a system's performance against the experts' expectations. For this purpose, the TURING test approach will rise again like a phoenix from the ashes. Although the test doesn't say

[1] Similarly, it is quite unlikely that airplanes flying like birds, ships swimming like dolphins, e.g., and trains behaving like a herd (or, more moderate, like a caravan) of camels are considered a success of science and technology.

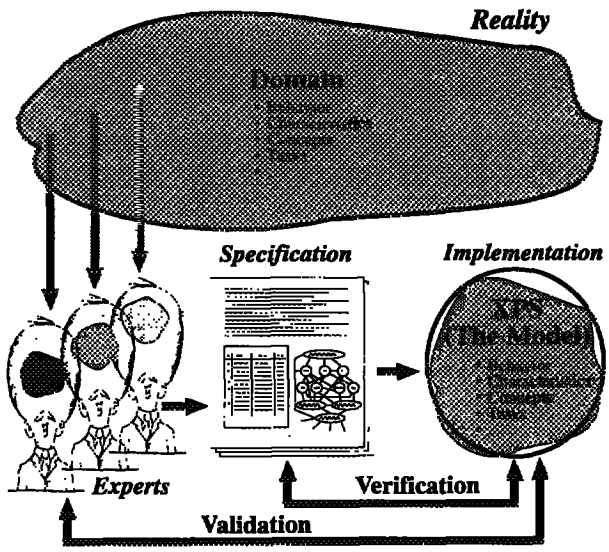


Figure 2: Relating Validation and Verification Taking Experts as the Basic Knowledge Source Into Account

anything about a system's "intelligence", it says something about a system's validity with respect to expert's knowledge.

## Basic Formalizations

In the minimal formal setting assumed so far, there are only two sets $I$ and $O$. On these sets, there is somehow given a target relation $\mathcal{R} \subseteq I \times O$. Under the assumption of some topology on $I$, one may assume that $\mathcal{R}$ is decomposable into functional components $\mathcal{R}_i^y$, where $y$ ranges over some subset of $O$ and $i$ ranges over some index set, such that

1. $\mathcal{R} = \bigcup \mathcal{R}_i^y$,
2. $\mathcal{R}_i^y \subseteq I \times \{y\}$ (for all $i$ and $y$),
3. the number of $\mathcal{R}_i^y$ is finite,
4. the cardinality of $\mathcal{R}$ (and therefore the cardinalities of $\mathcal{R}_i^y$) is (are) finite, and
5. every $\mathcal{R}_i^y$ is convex.

Although these conditions seem rather natural, for our considerations the conditions 1 and 2 are sufficient.

There are some elementary requirements to characterise expertise:

1. On the one hand, an expert's knowledge should be *consistent* with the target phenomenon.
2. On the other hand, everybody who is not providing any response at all is always consistent. Thus, one needs another requirement of expertise to comlement consistency, which is *completeness*.

Informally, from the possibly large amount of correct answers to an admissible question, an expert should know at least one.

A certain expert's knowledge $\mathcal{E}_i$ about some target phenomenon $\mathcal{R}$ is assumed to be a particular relation $\mathcal{E}_i \subseteq I \times O$ such that the following requirements of expertise are satisfied[2]:

$\mathcal{E}_i \subseteq \mathcal{R}$ **[Exp1]**

$\pi_{inp}(\mathcal{E}_i) = \pi_{inp}(\mathcal{R})$ **[Exp2]**

Ideally, $\mathcal{E}_i$ contains exactly the target relation:

$\mathcal{E}_i = \mathcal{R}$ **[Omn]**

In some sense, **[Exp1]** is a condition of consistency, **[Exp2]** is a condition of completeness, and **[Omn]** is a property called omniscience. An expertise $\mathcal{E}_i$ is said to be competent, exactly if it is complete and consistent:

- *competence = consistency + completeness*

Thus, a team of $n$ experts with their domain knowledge $\mathcal{E}_1, \ldots, \mathcal{E}_n$ is said to be

- competent, exactly if it meets
$$\bigcup_{i=1}^{n} \mathcal{E}_i \subseteq \mathcal{R} \quad \text{and} \quad \pi_{inp}(\bigcup_{i=1}^{n} \mathcal{E}_i) = \pi_{inp}(\mathcal{R})$$
- omniscient, exactly if it meets
$$\bigcup_{i=1}^{n} \mathcal{E}_i = \mathcal{R}$$

It might be usually unrealistic that one tries to find a competent team of experts. Vice versa, every team of experts is implicitly determining its own area of competence. In the practical world, because of not having a direct access to $\mathcal{R}$, we estimate $\mathcal{R}$ by $\bigcup_{i=1}^{n} \mathcal{E}_i$. That is, because we can't speak about formally "grey arrows" (cf. figure 2 above).

## Systematic System Validation

Based on the minimal formalisms provided, we are now able to develop our focused validation scenario.

- There is assumed an (implicitly given) desired target behavior $\mathcal{R} \subseteq I \times O$.
- There is a team of $n$ experts which is considered to be omniscient.
- There is some system to be validated. Its input / output relation is called $\mathcal{S}$.

Ideally, a system is omniscient, i.e. it meets

- $\mathcal{S} = \mathcal{R}$ **[S-Omn]**

Practically, a user may be satisfied with a system which is competent, i.e. which meets the requirements of consistency and completeness:

- $\mathcal{S} \subseteq \mathcal{R}$ **[S-Cons]**
- $\pi_{inp}(\mathcal{S}) = \pi_{inp}(\mathcal{R})$ **[S-Compl]**

But these three properties relating a given system's behavior directly to the target phenomenon are usually undecidable or, at least, unfeasible. Figure 3 is incorporating the minimal setting of formal concepts developed within the present paper.

[2]For notational convenience, we have introduced $\pi_{inp}$ to denote the projection of (sets of) pairs from $I \times O$ to the first argument, which is the input-part.

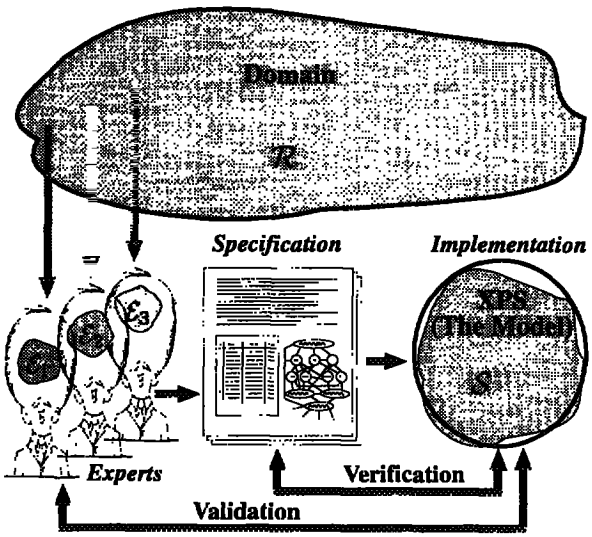


Figure 3: Relating Validation and Verification Based on a Minimal Setting of Formalized Human Expertise

The TURING test approach deals with a methodology to systematically provide some evidence for **[S-Cons]** and **[S-Compl]** and is based on the following intuition.

1. Some team of experts meets at least **[Exp1]** and **[Exp2]**, ideally, it meets **[Omn]**.
2. It's desirable, but unprovable, whether or not some system meets the corresponding conditions **[S-Cons]** and **[S-Compl]**.
3. If the system, when systematically interrogated, behaves like some expert (who satisfies **[Exp1]** and **[Exp2]**), this is understood as some strong indication that the system satisfies **[S-Cons]** and **[S-Compl]**, correspondingly.

Intuitively, nothing better could be imagined. In general, there are no proofs of properties involving $\mathcal{R}$, like **[S-Cons]** and **[S-Compl]**.

The research undertaken concentrates on the derivation of suitable interrogation strategies. Especially, one needs to know "what questions to ask" and "what to do with the answers". This is the question for suitable test sets and for an evaluation methodology of the interrogation results.

## Essential Problems Beyond the Basic TURING Test Methodology

The simplicity of the approach above allows a lucid introduction of fundamental properties like consistency, completeness, and competence. These concepts occur in more complex settings as well.

The present section is intended to systematically introduce a few basic generalizations of the elementary approach above. Every issue is addressed by only a very short introduction which should be understood a launching pad for future work.

### Context Conditions

Within some relation $\mathcal{R}$, knowing or believing some $(x, y) \in \mathcal{R}$ might be usually coupled to know resp. believe some $(x', y') \in \mathcal{R}$, as well. Analogously, some $(x, y) \in \mathcal{E}_i$ might rule out some other knowledge $(x', y')$ from the expert's perspective.

It seems promising to express context conditions by similarity concepts. Assume any binary function $\sigma$ called a similarity measure with $\sigma : (I \times O) \times (I \times O) \rightarrow [0, 1]$. In the simplest case, $\sigma$ may be binary. A similarity measure can be used to rule out incompatibilities from an experts knowledge.

$\sigma((x, y), (x', y')) = 0 \implies$
$\neg(\ (x, y) \in \mathcal{E}_i \wedge (x', y') \in \mathcal{E}_i\ )$ **[Incomp]**

This approach may be refined immediately, if one replaces the strong assumption $\sigma((x, y), (x', y')) = 0$ by the more moderate version $\sigma((x, y), (x', y')) < \varepsilon$ for some $\varepsilon$. Even more important, this suggests a dual version

$\sigma((x, y), (x', y')) > \eta \implies$
$(x, y) \in \mathcal{E}_i \Rightarrow (x', y') \in \mathcal{E}_i$ **[$\eta$-Coh]**

expressing some coherence of expert's knowledge. Informally, **[Incomp]** expresses that two knowledge pieces which are extraordinarily unsimilar (perhaps, even contradictory) can not simultaneously belong to one expert's knowledge. In contrast, the condition **[$\eta$-Coh]** of $\eta$-coherence states that knowing one fact implies the knowledge of every other one provided the similarity of them exceeds $\eta$.

### Validation in Open Loops

Essentially, the use of an intelligent system takes normally place in an open loop of environment (resp. human) - machine interactions. There is usually no way to estimate the number of interactions sufficiently precise. Thus, the following formalization seems reasonably simple and adequate.

Instead of $\mathcal{R}$, some target behaviour $\mathcal{B} \subseteq (I \times O)^*$ contains infinite sequences of action-response pairs. Even more specific, we formally assume $\mathcal{B} \subseteq (I \times O)^+$ to exclude the empty interaction sequence $\lambda$. As man-machine interaction will usually last only for a finite time, one need to reason about initial segments of those infinite sequences. By $\sqsubseteq$ we denote the prefix relation among words which is canonically extended to the case that the second argument belongs to $\mathcal{B}$. Furthermore, $A \sqsubseteq B$ abbreviates the case that for every element $a \in A$ there is some element $b \in B$ such that $a \sqsubseteq b$ holds. Finally, we need an generalization of the projection concept. Both for finite and for infinite sequences $s = (x_1, y_1) \ldots (x_n, y_n)$ resp. $s = (x_1, y_1), (x_n, y_n), \ldots$, the term $\pi_{inp}(s)$ abbreviates $\{x_1, x_2, \ldots\}$, accordingly. $\pi_{inp}$ is extended to sets via $\pi_{inp}(S) = \bigcup_{s \in S} \pi_{inp}(s)$, as usual. Expert activities are assumed to be finite, i.e.

$\mathcal{E} \subseteq \bigcup_{i=1}^{\infty} (I \times O)^i$ **[Fin]**

Based on this extension, consistency and completeness can be rewritten as follows.

$\mathcal{E} \sqsubseteq \mathcal{B}$ **[Exp*1]**

$\pi_{inp}(\mathcal{E}) = \pi_{inp}(\mathcal{B})$ **[Exp*2]**

As before, teams of experts are of interest. Here, we refrain from rewriting the other formalizations discussed in the previous section.

### Vagueness and Uncertainty

There is an intrinsic fuzzyness of the approach to validate intelligent systems against the opinion of some team of human experts. If a subject is difficult, humans rarely agree completely. This applies in particular to those complex application domains where AI systems are intended to work.

Even under the assumption of consistency and completeness of experts' knowledge it might happen that experts do not agree on several pairs of input and system's response. This is due to the fact that - in formal terms - the symmetric difference of some $\mathcal{E}_i$ and some $\mathcal{E}_j$ might be non-empty[3].

From the TURING test perspective, to cope with vagueness and uncertainty in the experts' knowledge, one might arrange an interrogation setting in which statistical results are evaluated (cf. (Knauf, Gonzalez, and Philippow 1997)).

### Improvements by Learning

Experimentation with a system under inspection may lead to the discovery of knowledge outside the competence of the available team of experts, i.e. cases in $\mathcal{R} \setminus \mathcal{E}$. Systematic knowledge discovery of this type requires a careful design of experiments as well as some methodology for analyzing the experimental results.

This idea is essentially based on the insight that system validation understood as the process of determining validity proceeds in steps over time. During this process, one might work towards improving the system's validity for passing the TURING test more successfully. The system is learning when being examined.

## Conclusions

The TURING test turns out to be an appropriate metaphor for intelligent systems validation.

System validation cannot guarantee, in general, that a system is the right one, i.e. that is is appropriate to solve all problems of a certain target class. The best system validation can certify is that a given system is at least as competent as some specified team of human experts. Thus, any TURING test result can only be interpreted with respect to the scenario, i.e. the involved experts, the asked questions, and the way to evaluate the answers.

Last but not least, there is obviously an urgent need to elaborate the concept of validity statements. The authors are convinced that the only way to use any TURING test result for system refinement is to develop validity statements which are somehow structured. Those statements should contain information about the part of knowledge, which is invalid, and about the kind of invalidity. Any validity statement, which is just an average degree of validity on any given linear scale can't achieve that.

[3]Loosely speaking, the symmetric difference of two experts' knowledge is all what the one knows, but not the other.

## Acknowledgements

The second author's work has been substantially supported by the Meme Media Laboratory of Hokkaido University, Sapporo, Japan. Furthermore, the authors gratefully acknowledge the sponsoring by the German Research Fund (DFG) within the project on Expert System Validation under grant Ph 49/3-1.

Last but not least, the authors wish to express their gratitude to all members of VAIS, the international and interdisciplinary Research Institute for Validation of AI Systems[4]. Communications among the team members of the institute have always been very stimulating and influential.

## References

Abel, Th.; Gonzalez, A. 1997. Utilizing Criteria to Reduce a Set of Test Cases for Expert System Validation. In (Dankel 1997), 402-406

Abel, Th.; Knauf, R.; Gonzalez, A. 1996. Generation of a minimal set of test cases that is functionally equivalent to an exhaustive set, for use in knowledge-based system validation. In (Stewman 1996), 280-284

Dankel, D.D. ed. 1997. Proc. of the 10th Florida AI Research Symposium (FLAIRS-97). Daytona Beach, FL: Florida AI Research Society

Halpern, M. 1987. Turing's test and the ideology of artificial intelligence. *Artificial Intelligence Review* 1:79-93.

Herrmann, J.; Jantke, K.P.; Knauf, R. 1997. Using structural knowledge for system validation. In (Dankel 1997), 82-86

Knauf, R.; Gonzalez, A.; Philippow, I. 1997. Towards an Assessment of an AI Sytems's Validity by a Turing Test. In (Dankel 1997), 397-401

O'Keefe, R.M.; O'Leary, D.E. 1993. Expert system verification and validation: A survey and tutorial. *Artificial Intelligence Review* 7:3-42.

Stewman, J.H. ed. 1996. Proc. of the 9th Florida AI Research Symposium (FLAIRS-96). Key West, FL: Florida AI Research Society

Turing, A.M. 1950. Computing machinery and intelligence. *Mind* LIX(236):433-460.

[4]cf. the internet appearance of the institute: http://kiwi.theoinf.tu-ilmenau.de/vais/