Validation Method for Intelligent Systems

Setuo Tsuruta* , Takashi Onoyama**, Sen Kubota** and Kazuko Oyanagi**
* Systems Development Laboratory, Hitachi, Ltd.
1099 Ohzenji Asao-ku Kawasaki-shi, 215-0013, Japan
e-mail: tsuruta@sdl.hitachi.co.jp
** Research & Development Department
Hitachi Software Engineering Co., Ltd.
6-81,Onoe-cho,Naka-ku,Yokohama,231-0015,Japan
e-mail: {ono, kubosen, oyanagi} @ori.hitachi-sk.co.jp

Abstract

Knowledge incorporated intelligent solving methods are prevailing in practical planning and scheduling because of the large problem size and complex constraints. However, when intelligent methods are used in real applications, it is an important problem to validate that both optimality and response time are at expert level. Turing Test approaches are useful in validating intelligent systems. Nevertheless, validation burdens experts with heavy tasks. In distribution route scheduling, for instance, validation load is considerable since patterns of distribution spots are numerous. A bi-directional, many-sided explanation typed multi-step validation method, which enables to share the validation tasks among experts, KEs and computers, is proposed to diminish the load on busy experts, for validating intelligent systems.

1. Introduction

Theoretically, distribution route scheduling can be classified as a traveling salesman problem (TSP). However, practical distribution route scheduling problems are extremely complex due to their scale and constraints. These problems are solved by intelligent and approximate solving methods, which utilize various knowledge and heuristics (Kubo 1994; Kubota et al. 1999). Now, one problem we face in applying these intelligent solving methods to real applications is to validate that both their solution optimality and response time are always at expert level.

Ordinary software validation methods (Myers 1979) have made considerable progress through focusing on detection of programming bugs. Validation methods of expert systems (Terano 1992) have been proposed. However, they proposed almost nothing more than general check items. The Turing Test approaches (Turing 1950; Knauf et al. 1998; Knauf, Gonzalez and Jantke 1999) seem promising in validating intelligent systems, but they impose excessively heavy burdens and responsibility on experts.

Experts are very busy and anxious about losing their vocational superiority by the appearance of intelligent systems. In order to lessen the validation load on experts and make their cooperation smooth, a validation method is proposed. By utilizing the system, experts, KEs and computers can share the validation load.

This validation method is proposed on the basis of our experiences in distribution scheduling problems, but the method can be generally used in validation of practical intelligent systems.

Chapter 2 describes problems encountered in validating distribution route scheduling method. Chapter 3 describes the proposed validation method. Chapter 4 compares this method with other methods and makes concluding remarks.

2. Problems in Validating Distribution Scheduling Methods

2.1 Characteristics of Validating Distribution Route Scheduling Methods

We have been building various distribution planning systems and distribution route scheduling systems that include a knowledge base and/or algorithms, for organizing optimal distribution logistics networks (Kubota et al. 1999). For example, these distribution logistics networks handle transportation of products and parts from several hundreds of suppliers to several dozens of factories and deposits all within these networks. The characteristics of these solving methods and validation method are as below:

1) Intelligent solving methods

Because of its scale, practical distribution route scheduling requires intelligent solving methods or high level approximate solving methods which use heuristics, know-how, and rules of thumb to ensure both their response time and solution optimality. When these solving methods are used in real application products, the solution optimality, response time, and their stability must be guaranteed from the viewpoint of product liability. However, input patterns, which fail to guarantee these items, can be created relatively easily in many intelligent solving methods utilizing heuristics and knowledge. Solution optimality and response time are occasionally quite different depending on problem patterns. Needless to say, test patterns have to be selected, taking test time and economy (Myers 1979) into consideration. Anyhow, product liability requires validation of a comparatively large number of test patterns in validating these intelligent solving methods. In order to decrease test patterns and guarantee the validity of methods for other patterns, tests conducted only in a black-box style will be inadequate. This is because the output of each component method or its convergence state has to be investigated in order to supplement omitted test patterns through checking mutual relationships among those component methods.

2) Busy experts in a delicate social position

Checks by experts are indispensable in validating distribution route scheduling methods. However, as mentioned above, a large number of cases have to be tested for validation. Time and labor needed to make responsible judgment on such numerous test cases are an unbearable burden for experts. Since they are already heavily loaded with daily work, their cooperation cannot be readily obtained. Their real intention is that they do not want to cooperate. This is because they are in such a delicate social position as to lose their vocational superiority with very high probability, if high level intelligent solving methods become available on the market as products (Tsuruta et al. 1997).

It will be all the more difficult to obtain the cooperation of experts if, even anonymously, the results of their judgments are inputted in the computer and are compared with others (Salecker and Knauf 1999). This reason lies in the possibility of their weaknesses being exposed, their occupational judgment being interfered and their vocational superiority being undermined. In case an expert's solution was judged useless by KEs and it was reported to the expert, we can easily imagine that the expert placed in above-mentioned delicate position would become reluctant to cooperate in subsequent validation. On the other hand, there are many experts who never accept solutions obtained by methods different from theirs. On the other hand, in many cases, even experts cannot make responsible judgments unless explanations (such as why the results are good or bad) are given concerning the method, its component methods, each know-how, utilized knowledge, and their effects (Tsuruta et al. 1997).

3) Possibility of sharing validation task

Indeed, automatically created routes have to be checked out in order to decide whether these routes can be actually useful. But, even KEs can check more than the half of these routes through their shapes. Furthermore, the results such as routes' lengths and tour time can be represented as numerical values. Thus, through comparison with correct solutions in benchmark tests or solutions obtained by other methods, there is also high possibility of mechanical validation or at least mechanical detection of the inferiority of methods or systems.

2.2 Problems in Validation

The following problems occurred in our validation of distribution route scheduling methods and systems.

1) Necessity of KEs' pre-validation and many-sided explanation

Solving methods usually contain component methods and parameters. The combination of computer methods and the adjustment of the parameter in each component method are important. However, the combination of each component method including parameters in each method was so complex that it caused errors in test procedures and in data collection.

For example, random restart strategy required sufficient number of repetitive executions of component methods and randomness of the initial solutions so that it can be practically applied to such a case as scheduling, which requires real-time man-machine interaction. The use of heuristics such as the NI (Nearest Insertion) method in the initial construction was effective to increase the above number of repetitive executions. On the contrary, this decreased the randomness of the initial solutions and increased the potentiality of resulting in a local minimum. Using only the final output results of solving methods failed to validate this trade-off between the two. Collection and analysis of resultant data at a halfway stage to check randomness in the initial construction methods and convergence status in improvement methods were necessary. However, these data were not collected systematically or in advance.

2) Necessity of explanation and mutual communications

Even experts would have difficulty in their responsible validation of real products if only the final output results were available. They had such difficulty especially when the results were different from those that would have been generated using the experts' own method or when a problem pattern such as the location pattern of distributing spots was entirely unfamiliar to them. Thus, to conduct responsible validation, they need to know problem solving process and how the resultant solution was obtained. Because of this, experts frequently requested explanations to computer specialists such as KEs and SEs. With each inquiry, they tested for comparison with other solving methods, and thus repeated the same tests many times, collected data again and again, and edited reports to explain the foregoing solving processes and reasons. These tasks required a long time and man-hours.

What is worse, many problems occurred. For example, they often failed either in repeating the completely same tests as is needed for the explanations, or in collecting necessary data in these tests. Therefore, busy experts were asked to cooperate with the tests of the same items many times.

3. Validation Method

3.1 Concept of Validation Method

In order to solve the foregoing problems, a bi-directional many-sided explanation typed multi-step validation method (see fig.1) is proposed as follows:

1) Multiple step validation

In this method, experts, KEs (possibly including SEs and other information system specialists) and computers can share validation tasks and cooperate to enhance validation efficiency. Validation is performed in three steps: (1) automatic validation by a computer, (2) validation by KEs and (3) validation by experts. Among many test cases, experts have only to check test cases that neither KEs nor computers can judge correctly. Thus, validation load on busy experts can be greatly decreased.

2) Many-sided explanation

To avoid repeating the same tests, computers can automatically collect, edit, and store various data necessary for each validation of the above multiple validation steps, in advance. These data comprise resultant data and data specially collected or processed for many-sided explanations. The latter include: (1) statistical tables of each resultant data of a solving method and its component methods, (2) explanations of problem solving process such as convergence status tracing graphs and tour (or traveling route) maps of the initial and halfway solutions.

Many-sided explanations or data can be provided in various modes depending on the needs in each validation step, and are stored separately for KEs and for experts. For example, the above tour maps are provided for KEs together with topological or relatively reduced road maps, but they are provided for experts together with less reduced road maps.

3) Adding explanations by KEs and experts

In addition to automatic explanations, KEs and experts can add their own explanations such as scratches or comments in their validation steps for themselves and for others. These explanations are called comments. Comments are described in natural language. Since comment is an item of a validation result, they are referenced together with each validation result in the below-mentioned validation case base. Especially, KEs can add explanations such as comments to the final output of a system or method as well as the output in each execution stage of its component methods. These comments include explanations of reasons why the solution was obtained.

These comments are added also separately for experts and for KEs, through specifying recipients. Conversely, each ID of commentators, who added explanations, is automatically tagged to each explanation.

4) Bi-directional and parallel validation on the Web

Bi-directional interactive and parallel validation is supported, using the Internet or Intranet. Experts can lessen their validation load through checking only test cases requested by KEs, using the results and explanation data on the Web. Meanwhile, experts can efficiently enhance the dependability of validation by asking questions or inquiring information to KEs or computers, and obtaining answers from them, interactively through bi-directional many-sided explanations and comments on the Web. Similarly, multiple experts and even multiple KEs can check allotted and shared test cases in parallel through bi-directional many-sided explanations on the Web.

The validation results of experts and KEs are stored in the database (Test DB in fig.1) as a validation case base, and used in editing validation reports. Yet, KEs and automatic validation computers can reuse this case base for the subsequent validation.

Thus, efficient, prompt, dependable, and load sharing validation can be realized.

3.2 Validation Steps

Fig. 1 shows our proposed validation method. And the steps of this validation method are described below. In this method, program-level tests by programmers are assumed to have finished.

1) Automatic validation

Triggered by the test manager who is the leader among KEs or in the test team, the validation programs of computers automatically perform the following tasks in accordance with the test requirements specified by the test manager. First, all combinations of solving methods and parameters to be tested and necessary test cases are fetched from the test database and a test is performed. Next, the resultant data are converted into statistical values, and they are then determined whether or not they satisfy the estimation criteria set in advance. The resultant data are presented to KEs, including information system specialists such as SEs and system analysts, by adding raw data, statistical data, automatic evaluation results (overall and individual), and explanations of problem items provided separately for KEs and for experts.

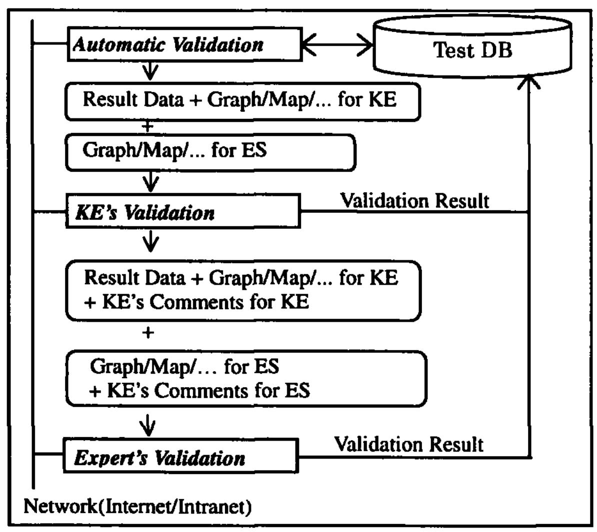

Figure 1. Bi-directional Many-sided Explanation Typed Multi-step Validation Method

2) Validation by KEs

KEs examine errors found in automatic validation as to whether they are errors by programming bugs, by the solving methods or component method, or by the test method, using explanations for KEs (including those of halfway status conducted by a component method) as references. Test methods' errors include parameter-setting errors. If a programming bug is suspected, the data is returned to a program-level test and is removed from the target of this validation. The test results passed through automatic validation are checked by KEs at least on a sampling basis. If a problem is found, KEs change the results of automatic decision and make further examinations. The results are sent to experts by KEs when responsible validation by KEs alone is judged impossible even if the results are on the borderline or even close to an optimal solution and believed to be correct. KEs then add comments to the explanations for validation by experts.

KEs utilize many-sided explanation for each stage of solving methods and component methods, in order to understand the behaviors of solving methods and to judge the correctness of the test results. Problematic resultant data are returned to the automatic validation step, using the bi-directional interaction environment. If necessary, similar test data can be used for revalidation. By presenting the results of comparison test between the solving method to be validated and other solving methods, KEs can judge the optimality and stability limits and can add his comments to the explanations for KEs and experts.

KEs attaches all knowledge obtained in this bi-directional validation task to the above-mentioned comments, to be used as validation information by KEs and experts. Comments can be hierarchically classified into such groups as solving method's comments and each component methods' comment group.

3) Validation by experts

Experts make judgment only on the test results for which validation has been requested by KEs. Besides using test results on the Web, experts validate also using statements, diagrams, charts and maps presented by the validation tool as explanations for experts, and also using comments added by KEs to the test results. Experts make inquiries regarding doubts and additional information to KEs and SEs. Experts can return data to automatic validation step for revalidation as is mentioned in 2).

In order for busy experts to readily input, the validation results of experts are an evaluation value described as a number of 3 through 5 grades, and a short comment in natural language. They are saved as a validation case base for use in a comprehensive report of validation. The results will also be reused in future validation as validation know-how.

4) Aggregation step

All validation results are automatically aggregated and documented. If a problem is found, the test manager asks for explanation to the expert or the KE who made the decision and prepares for a final comprehensive report.

4. Comparative Evaluation and Conclusions

4.1 Other Validation Methods and Tools for AI Systems

Many excellent theories and ideas based on the well-known Turing Test (Turing 1950) have been proposed as methods to validate AI systems in order to determine whether or not problem-solving knowledge (solving methods) is at expert level. All these methods validate in a black-box style merely examining inputs and outputs (Abel and Gonzalez 1997; Knauf and Gonzalez 1997; Knauf et al. 1998; Knauf et al. 1999; Salecker and Knauf 1999). Compared with our proposed method, the following problems are conceivable with these methods:

1) Validation load on experts

Differences in solution optimality and response time in intelligent solving methods, which use heuristics and knowledge, are large depending on the problem pattern. In terms of product liability, validation of their stability is mandatory. However, the number of test cases tends to increase to a significantly large number in black-box style tests. The idea of considering "cases" as test data is interesting as a method for reducing the number of test cases in case based system (Knauf, Gonzalez and Jantke 1999). In distribution route scheduling, however, even slight differences in traveling spots frequently cause great differences in solutions. For this reason, it is difficult to apply case based approaches to distribution route scheduling problems. The method of reducing the number of test cases using a formally described domain model and evaluation criteria is also interesting (Abel and Gonzalez 1997). One problem with this method, however, is that man-hours are needed to create a formal model and evaluation criteria so that it cannot be easily used in practical applications. Normally, test cases are narrowed down in accordance with informal specifications and evaluation criteria. Even in this case, as product liability is involved, these intelligent solving methods must be validated with relatively a very large number of test patterns and loads on experts are critical. Furthermore, black-box style tests alone are not adequate, as operations of patterns other than test patterns must be also guaranteed. Turing-Test-like tools through a network in order for experts to anonymously cooperate in validation tasks have also been proposed (Salecker and Knauf 1999). Nevertheless, these methods do not diminish the foregoing validation load problems.

In many cases, experts cannot make responsible validation by examining merely inputs and outputs, or unless the problems can be easily analogized on the basis of their own experiences. What is worse, such validation burdens busy experts with heavy as well as mental load.

These problems remain, even with the methods and tools of validation, in which experts solve the same problem and rate their results each other anonymously, though they are interesting (Knauf and Gonzalez 1997; Salecker and Knauf 1999).

2) Necessity of Many-sided Explanation

Solutions more optimum and higher in quality than the ones obtained by experts can also exist. For example, in distribution route scheduling to build a logistics distribution network, frequently there occur problems (traveling spot patterns) which experts encounter for the first time. Regarding solutions of problems, which experts first encounter, and highly optimal solutions to win over competitions, validation through merely examining inputs and outputs like Turing Test (Turing 1950; Knauf et al. 1998), is difficult even for experts, especially when product liability should be considered. There exists a necessity of various many-sided explanations to each stage's solution behavior like halfway convergence status of solutions and component methods.

4.2 Conclusions

A bi-directional many-sided explanation type multi-step validation method has been proposed to diminish the validation load on experts, who are busy and stand to lose their vocational superiority by the application of intelligent expert systems, and to make their cooperation smooth. By utilizing this system, experts can concentrate on the areas that cannot be covered by the computers or KEs and can make responsible judgments by utilizing bi-directional, multi-step and many-sided explanation types.

This validation method has been developed tailored to distribution scheduling and other purposes. This method ensures efficient validation due to the decrease of load on experts by sharing validation tasks through multi-step validation and by many-sided explanation including explanations at halfway stages of solving methods. From this viewpoint, the method can be used in validating general intelligent solving methods of scheduling problems, as well as general intelligent solving methods and intelligent systems.

References

Abel, T.; and Gonzalez, A.J. 1997. Enlarging a Second Bottleneck: A Criteria-based Approach to Manage Expert System Validation Based on Test Cases. In Proceedings of the 10th International Florida AI Research Society Conference, Daytona Beach, FL.

Knauf, R.; and Gonzalez, A.J. 1997. Estimating an AI system's validity by a TURING Test. In Proceedings of 42nd International Scientific Colloquium. Technical Univ. of Ilumenau.

Knauf, R.; Jantke, K. P.; Abel, T., and Philippow, I. 1998. Fundamentals of a TURING Test Approach to Validation of AI Systems. In Proceedings of 42nd International Scientific Colloquium, Technical Univ. of Ilumenau.

Knauf, R.; Gonzalez, A.J.; and Jantke, K. 1999. Towards Validation of Case-Based Systems. In Proceedings of 12th International Florida AI Research Society Conference, Orland, FL.

Kubo, M. 1994. Invitation to Traveler salesman problem. Communications of the Operations Research Society of Japan, 21(2):91-96.

Kubota, S.; Onoyama, T.; Oyanagi, K.; and Tsuruta, S. 1999. Traveling Salesman Solving Method fit for Interactive Repetitive Simulation of Large-scale Distribution Network. In Proceeding of the 1999 IEEE SMC Conference, Tokyo.

Myers, G.J. 1979. The Art of Software Testing. John Wiley & Sons.Inc.

Salecker, D.; and Knauf R. 1999. Validation Manager A Tool for AI System's Evaluation by A Turing Test-Like Methodology, In Proceeding of the 12th International Florida AI Research Society Conference, Orland, FL.:AAAI Pres.

Terano, T. 1992. Expert System Evaluation Manual, Tokyo:Ohmsha.

Tsuruta, S.; Eguchi, T.; Yanai, S.; and Ooshima, T. 1997. Dynamic Goal centered Coordination AI system: Application and the validation problem, In Proceeding of 42nd International Scientific Colloquium, Vol.2, 52-58.Technical Univ. of Ilumenau.

Turing. A. M. 1950. Computing machinery and intelligence, Mind, 59(236):433-460.