Reorganizing TRIZ Solution Generation Methods into Simple Five in USIT

**Toru Nakagawa (Osaka Gakuin Univ.),
Hideaki Kosha (Fuji Photo Film Co., Ltd., Japan) and
Yuji Mihara (Fuji Photo Film Co., Ltd., Japan)**

This paper was presented in ETRIA World Conference: TRIZ Future 2002, Strasbourg, France, Nov. 6-8, 2002, pp. 333-345, Japanese translation posted on "TRIZ Home Page in Japan", Sept 2002; English version posted on "TRIZ Home page in Japan", Nov. 2002

Abstract

As Solution Generation methods, TRIZ has provided a large number of techniques and principles: 40 Principles of Invention, 76 Standards of Inventive Solutions, Trends of Evolution of Technological Systems, Separation Principle, etc. This shows the richness in TRIZ, but also makes pitfalls of difficulty and confusion. In the present study all these TRIZ methods are reclassified in the framework of USIT (Unified Structured Inventive Thinking). USIT has only five Solution Generation Methods: i.e. Object Pluralization, Attribute Dimensionality, Function Distribution, Solution Combination, and Solution Generalization Methods. It is remarkable that the huge variety of TRIZ methods are smoothly mapped onto these five USIT methods. The USIT Solution Generation Methods are now enhanced much with TRIZ-origin methods and have clear guidelines. Thus the present work has reorganized TRIZ into a much simpler yet more effective process for problem solving: namely, USIT.

1. INTRODUCTION

As a methodology for creative problem solving, TRIZ (i.e., 'Theory of Inventive Problem Solving' in Russian abbreviation) [1-5] provides a full range of features for freer/wider thinking, for retrieving scientific/technological knowledge, for analyzing problems, and for generating solutions. Talking of the solution generation methods alone, TRIZ offers a large number of techniques and principles: they include 40 Principles of Invention, 76 Standards of Inventive Solutions, Trends of Evolution of Technological Systems, Separation Principle, etc. This shows the richness and outstanding strength in TRIZ. However, such a huge body of methods and knowledge bases is not easy to understand and even confusing for learners of TRIZ. Thus, unfortunately, the penetration of the Russian-born TRIZ into the West since 1990s has been much slower than being anticipated by TRIZ enthusiasts [6].

An approach to simplifying TRIZ for easier industrial acceptance was started in Israel in 1980s by forming SIT ('Systematic Inventive Thinking' or 'Structured Inventive Thinking') [7,8]. Then in 1995, Ed Sickafus at Ford Motor Co. adopted SIT and developed it further into USIT ('Unified Structured Inventive Thinking') [9]. USIT has been introduced and refined in Japan since 1999 [10,11]. The present paper intends to demonstrate that the whole TRIZ solution generation methods can be utilized in USIT in a much simpler, unified, and effective way.

The whole process of problem solving can be characterized by three principal phases, i.e., Problem Definition phase, Problem Analysis phase and Solution Generation phase, according to USIT. During the first two phases we prepare in various ways only for the last phase in which we want to make some breakthroughs to actually obtain new, innovative conceptual ideas. In the present paper we focus on this final and crucial phase of Solution Generation.

In the present study, comprehensive principles and rules in the Solution Generation methods in TRIZ have been reclassified into the framework of USIT. As described in [11], USIT has only five Solution Generation Methods: they are Object Pluralization Method, Attribute Dimensionality Method, Function Distribution Method, Solution Combination Method, and Solution Generalization Method. These are based on the concept of "Objects-Attributes-Functions" and have been found successful in accepting such a large variety of individual principles/rules in TRIZ. Thus, without loosing USIT's original idea of simplicity, USIT has been enhanced much into an equivalent to the huge body of TRIZ Solution Generation methods.

The present paper describes how the reorganization of TRIZ was done, how the resultant Solution Generation Methods of USIT look like, and how they can be studied and applied for creative problem solving.

2. MAPPING TRIZ METHODS ONTO THE USIT FRAMEWORK

2.1 Basics of USIT as a New Framework

Just like any other technological system, USIT has evolved from its original form developed by Sickafus [9] to current Japanese version refined by Nakagawa [11]. The whole procedure of USIT is illustrated in the flowchart shown in Figure 1. Problem solvers may proceed step by step along this flowchart*1 with simple guidelines learnable at heart and without needing any handbook or software tool.

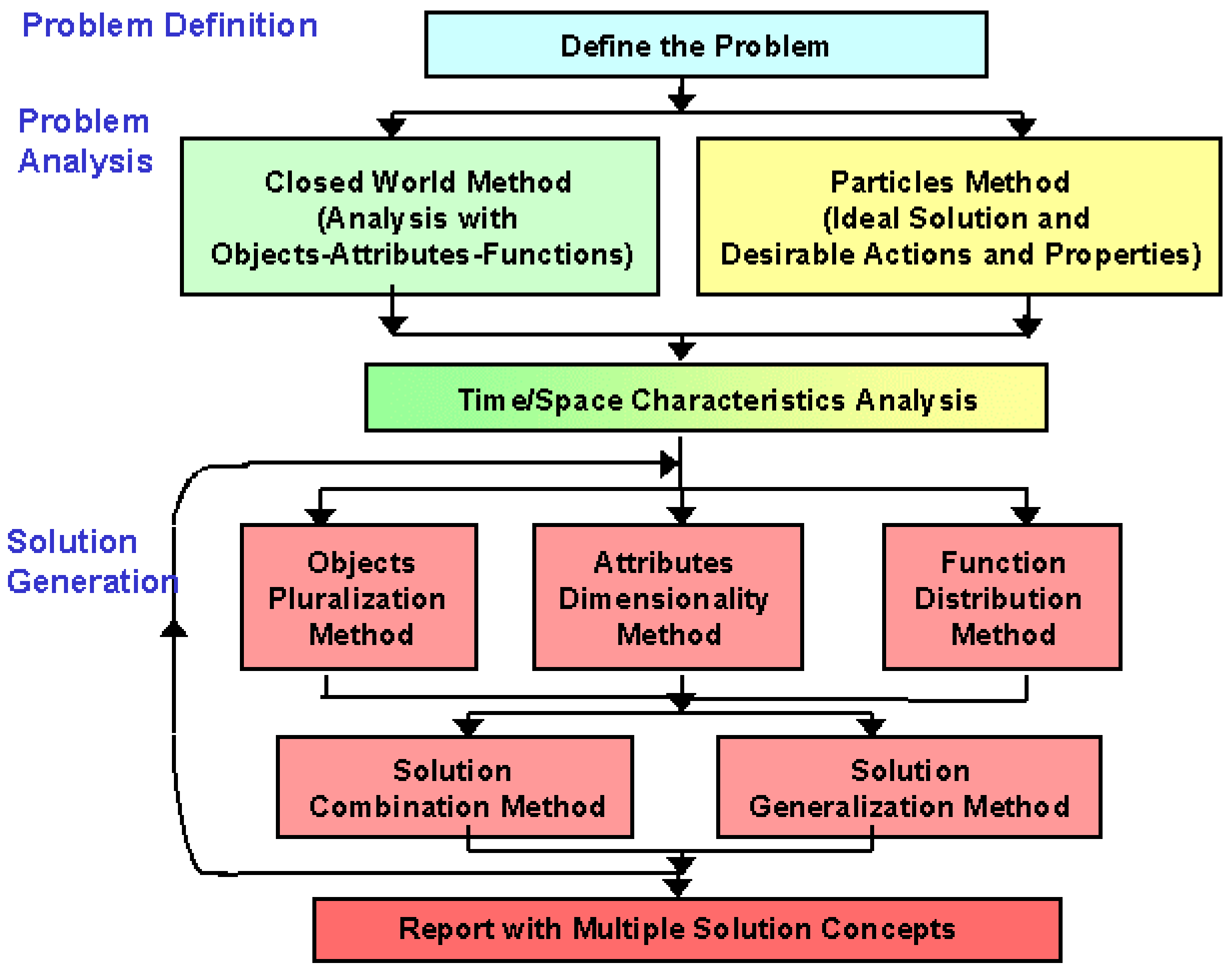

Figure 1: Flowchart of the Problem Solving Process in USIT

Note *1: Two parallel methods in the Problem Analysis Stage may be selected independently (i.e, in four patterns: A, B, A->B, and B->A). Five methods in the Solution Generation Stage may be used independently and repeatedly as many times as you want.

The five methods in the Solution Generation Stage, as described in [11], are:

1. **Object Pluralization Method:** Apply to (or operate on) every Object in the system so as to 'pluralize' it. 'Plural' here (as in the English sense) means any number except 1 (i.e., 0, 2, 3, ... inf., 1/2, 1/3, ... 1/inf., etc.)
2. **Attribute Dimensionality Method:** Apply to (or operate on) various Attributes of every Object in the system so as to modify the 'dimensionality' of the Object. I.e., activate/deactivate Attributes and vary Attributes in space and time.

3. **Function Distribution Method:** Apply to (or operate on) all the Fnctions in the system so as to 'distribute' (or rearrange) the functions among the Objects in the system (including newly introduced Objects). Also consider introducing new Functions.
4. **Solution Combination Method:** Apply to multiple solutions (or solution elements) and combine them in various ways in space, in time, in parts, etc. to form a new solution.
5. **Solution Generlaization Method:** Replace technical, specific terms in each solution with plain, generic terms.

The first three methods operate on Objects, Attributes, and Functions, respectively. In the preceeding stage of Problem Analysis, the problem solver should have fully analysed the system of the problem with the concept of 'Objects-Attributes-Functions'; this makes the application of these three methods smooth.

Some historical comments may be necessary on the fourth method especially for readers of the USIT textbook [9]: The 'Transduction' method by Sickafus has been resolved here into the Function Distribution Method (in the sense of energy-converting function) and the Solution Combination Method (in the sense of linking two functions). The 'Uniqueness' method by Sickafus has been moved mainly to the 'Space and Time Characteristic Analyis' method in the Problem Analysis Stage, and partially to the first four Solution Generation Methods listed above (in the sense of space- and time-characteristic solutions). The current fourth method, i.e. Solution Combination Method, was introduced in [11] so as to cover a wide range of solution techniques, which correspond to the ones supported by TRIZ Separation Principle.

The fifth method, Solution Generalization Method, intends to break Psychological Inertia, especially the one due to technical terms, and to systematize the solution search. The chapter 'Generification - A USIT Process' in Sickafus' textbook [9] is worth reading.

2.2 TRIZ Methods for the Solution Generation Phase

Since TRIZ is such a huge collection of findings, techniques, knowledge bases, examples, etc. for various phases/aspects of problem solving, we need some preliminary discussion on which we should include here as TRIZ methods for the Solution Generation phase.

The TRIZ textbook recently wrote by Darrell Mann [5] seems to be most suitable as the basis for this discussion. He has devided the (TRIZ) process of problem solving into the following four phases (as characterized by the tools listed together):

(I) Define: System Operator (9-Windows), Problem/ Opportunity Explorer, Function/ Attribute Analysis, S-Curve Analysis, and Ideal Final Result
(II) Select tool
(III) Problem solving: Technical Contradiction/Inventive Principles,

Physical Contradiction, S-Field Analysis/Inventive Standards, Trends of Technological Evolution, Resources, Kowledge/Effects, ARIZ, Trimming, Ideal Final Result, Psychological Inertia Tools, Subversion Analysis
(IV) Solution evaluation

In the sense of USIT, the Phase (III) should be divided into Problem Analysis and Solution Generation phases. We classified the tools in Phase (III) into the following four groups:

- Tools for Problem Analysis phase: Technical Contradiction (i.e., Contradiction Matrix), (formulation of) Physical Contradiction, S-Field Analysis, Psychological Inertia Tools (including Smart Little People), and Subversion Analysis
- Main tools for Solution Generation phase: Inventive Principles, Separation Principle (for solving Physical Contradiction), Inventive Standards, Trends of Technological Evolution (as the guiding principles), Trimming, and Ideal Final Result (as a guiding principle)
- Supporting knowledge tools for Solution Generation phase: Resources and Knowledge/ Effects
- Overall composite procedure (to be compared with USIT itself): ARIZ

Thus there are three most important methods (or collections of methods) we want to examine in the present work: they are Inventive Principles, Inventive Standards, and Trends of Technological Evolution. Three other methods (i.e., Separation Principle, Trimming, and Ideal Final Result) are rather standing alone and may be handled later individually.

Resources and Knowledge/Effects are not procedures in nature but rather knowledge which should be used to support the solution generation. Thus they may be set aside in the present mapping work.

Since USIT is proposed as an overall procedure for problem solving, it replaces ARIZ, and other similar overall procedures such as the ones recommended by Salamatov [2] and by Mann [5].

2.3 Sources of And Comments on TRIZ Solution Generation Methods

Following sources are used for the three main TRIZ Solution Generation methods:

- [P] **Inventive Principles:** Forty principles and their subprinciples (a, b, c, ...) taken from Salamatov[2]. (Lists in other references such as [1, 3-5] are essentially the same, with quite minor differences.) Cited here as [P15a], [P20abc], etc.
- [S] **Inventive Standards:** (So-called seventy-six) standard (solution guidelines) taken from Salamatov [2]. (Mann [5] regrouped them without changing the contents.) These Standards are cited here as [S1-2.1], [S1-2.1,2,3], etc. along with the second-level names, where the digits behind a dot stand for the third (and

fourth) level and the digits behind a comma shows the parallel citation for the third level.

- [T] **Trends of Evolution of Technological Systems:** 31 Trends taken from Mann [5]. Cited here as [T12], [T20], etc. with the title, where the numbers are attached in the order as appearing in [5].

The terminology in Inventive Principles and in Trends of Evolution are mostly understandable even for TRIZ beginners. One should note that the statements of Trends should be regarded not only as the observation but also as the guiding principle in our context.

Inventive Standards, on the other hand, are based on the S-Field Analysis formulation of the problem and use a terminology particular to TRIZ. Thus brief explanation may be appropriate:

In the S-Field Model (or SFM), the system of problem must be focused sharply by the representation with only two 'Substances' (S1 and S2) and one 'Field' (F). 'Substance' here is a generic term for representing 'something' and is essentially the same as 'Object' in USIT. 'Field' in TRIZ is a generic term particular to TRIZ for representing interactions, forces, energy, physical fields, functions, etc. In a well-defined system, the Substance S2 works as a tool by applying the interaction F onto the target Substance S1. The term 'Environment' stands for all other system components than S1 and S2 and also for anything existing or easily-accessible around the system. In various cases of problem system, either one of the three elements is missing/ inappropriate/ insufficient/ exceeding/ harmful/ etc. Inventive Standards specify the cases and state relevant solution guidelines as suggestions.

2.4 Heuristics Listed by Sickafus

In the USIT textbook [9], Sickafus listed 21 Heuristics in addition to detailed description of USIT Solution Generation Methods. They seem to reflect Sickafus' efforts for trying to merge TRIZ methods with his original ones in the style similar to Inventive Principles. Since they contain several unique viewpoints, we have decided to classify them also in the present framework.

Sickafus' Heuristics and their subcategories are numbered as they appear in [9], and are cited here as [H5a], [H5abc], etc. These Heuristics have emphasis on the viewpoints (e.g., 'Rate of events', 'Periodicity', etc.) and hence both positive and negative actions (e.g., introduce/remove, increase/decrease, etc.) are listed in their subcategories. Even though such reverse thinking is important to break Psychological Inertia, we think that our guidelines should show some most appropriate directions (just like in the statements of Trends of Evolution) while keeping the reverse thinking as a reminder applicable in general context.

2.5 Mapping, Regrouping, and Describing Processes

The three sources of TRIZ methods (i.e., Inventive Principles [P], Inventive Standards [S], and Trends of Evolution of Technological Systems [T]) and Sickafus' list of Heuristics [H] are handled at their lowest-level submethods such as [P1a], [S1-1.8.1], [T1], [H1a], etc. Each submethod is examined in its implication and mapped onto the USIT's five Solution Generation Methods. It should be noted that the mapping allows 1 to n and hence a TRIZ submethod can be mapped onto multiple places in the USIT methods. For instance, [P3c] of Local Quality Principle is mapped onto the Object, Function, and Combination methods, while [P3d] onto Attribute and Combination methods. In this manner, all the submethods of the four sources have individually found their appropriate candidate positions in the USIT framework.

Then all these (TRIZ-origin) submethods are regrouped in a hierarchical manner in the framework of USIT. Each of the five USIT Solution Generation Methods originally had a few submethods and brief guidelines [11]. The present work has urged to introduce a few more submethods and has much enhanced the contents in the form of lower-level submethods and their descriptions. To each USIT submethod, we have found contributions of sometimes all the four sources together, sometimes three, two, and one of them, and in some special cases none of them (that means being original in USIT).

Next (or , in actual, repeatedly in parallel to the regrouping work), we have described the guidelines of the USIT methods in a hierarchical manner so as to cover the intentions of the regrouped TRIZ-origin submethods. Even though the principles/techniques in TRIZ are stated in quite different levels of abstraction, we have tried to maintain the level of abstruction most suggestive as the hints for solution generation. It is important that the USIT submethod should not be restricted in its implication to the extents of the TRIZ-origin submethods mapped there. We should rather regard the latters as non-comprehensive examples of guidelines of the USIT submethods.

3. USIT SOLUTION GENERATION METHODS

3.1 Table of the USIT Solution Generation Methods

The USIT Solution Generation Methods thus obtained in the present work have the outline as shown in Table 1. The submethods marked with * are much enhanced or newly introduced in the present study as the result of merging TRIZ-origin submethods.

USIT Solution Generation Methods in their full extent are shown in the Appendix. Brief guidelines are attached at the levels of the five Methods and 32 submethods. Most of the submethods have some more detailed descriptions for application guidelines and for suggestive examples. At this level of each USIT submethod, a number of TRIZ-origin submethods are cited, including Inventive Principles [P], Inventive Standards [S], and Trends of Evolution [T].

A further expanded table of the USIT Solution Generation Methods, accompanied by the description of the TRIZ-origin submethods, will be published online in our 'TRIZ Home Page in Japan' [12]. With such a table one can access to the rich knowledge-bases of TRIZ methods and their cumulative examples, while staying in the clearer framework of USIT.

3.2 Merging TRIZ Minor Methods into USIT

In relation to the previous discussion in Section 2.2, we should comment here on how we handle several minor methods in TRIZ:

- **TRIZ Separation Principle:** This principle corresponds to the USIT method (4), i.e. Solution Combination Method. Separation Principle guides you to find some situation where the opposite simultaneous requirements in the Physical Contradiction can be separated, then to find partial solutions to fulfill the requirements separately, and finally to combine the partial solutions in a compatible way. Exactly for this third step, USIT Solution Combination Method gives you full guidelines. USIT way is much easier to understand and to apply. (Karasik [13] wrote on the history of Separation Principle and that the idea of separation in various dual views (other than time and space) was difficult to understand even for Altshuller.)
- **Trimming:** This corresponds to the USIT submethod (1a), i.e., 'Eliminate the Object'.
- **Ideal Final Result:** The 'Self-X' solution guidelines as described by Mann [14] is the main tool in this category for the Solution Generation phase. The guidelines should have their location in the USIT submethod (3h) ('Introduce/enhance the adapting/coordination/control Function'), together with the TRIZ [P25] ('Self-service').

Table 1: USIT Solution Generation Methods (Outline)

(1) Object Pluralization Method (1a) Eliminate the Object (into 0). (Simplification, Trimming) (1b) Multiply the Object (into 2, 3, ..., inf.). (1c) Divide the Object (into 1/2, 1/3, ..., 1/inf.). (1d) Unify multiple Objects into one. (1e)* Introduce a new/modified Object. (1f) Introduce an Object from the Environment. (1g)* Replace a solid Object with a powder/fluid/liquid/gaseous Object.

(2) Attribute Dimensionality Method

(2a) Deactivate/make irrelevant the harmful Attribute.
(2b)* Activate/involve a new useful Attribute.
(2c) Enhance the useful Attribute or suppress the harmful Attribute.
(2d) Introduce/enhance a spatial Attribute or distribute/vary in space a harmful/useful Attribute or Attribute's value.
(2e) Introduce/enhance a temporal Attribute or distribute/vary in time a harmful/useful Attribute or Attribute's value.
(2f)* Change the phase, utilize the phase change, or change the inner-structure of the Object.
(2g)* Utilize Attributes/properties at the micro level.
(2h)* Improve the properties/performance of the system as a whole.

(3) Function Distribution Method

(3a) Reassign the Function to a different Object.
(3b) Divide the compound/multiple Functions and assign them to different Objects or different parts of an Object.
(3c) Unify multiple Functions and assign the unified Function to an Object.
(3d)* Introduce a new Function and assign it to an Object.
(3e) Distribute/vary the Function in space or utilize the spatial distribution/motion/vibration Function.
(3f) Distribute/vary the Function in time.
(3g) Realize the detection/measurement Function.
(3h)* Introduce/enhance the adapting/coordination/control Function.
(3i)* Achieve the Function with a different physical principle.

(4) Solution Combination Method

(4a) Combine solutions functionally.
(4b) Combine solutions spatially.
(4c) Combine solutions temporally.
(4d) Combine solutions structurally.
(4e) Combine solutions at the principle level.
(4f)* Combine solutions at the super-system level.

(5) Solution Generalization Method

(5a) Generalize/specify the solution for associative thinking.
(5b) Construct a hierarchical system of solutions.

3.3 Notes on Illustrations of the USIT Solution Generation Methods

In Appendix, the USIT Solution Generation Methods at their submethod level are supplied with simple illustrations. We use the symbols such as shown in Figure 2.

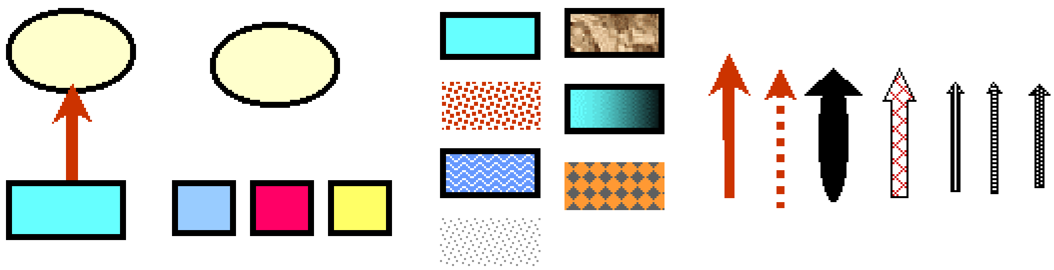

Figure 2: Symbols Used for Illustrating the USIT Solution Generation Methods.
Ovals: Target Objects; Rectangles: Objects (or Tool Objects);
Colors and Patterns in rectangles: Attributes; Arrows: Functions

The left-most part of Fig. 2 shows the basic system representation in USIT. The upper Object (most important in the system and hence shown at the top with an oval) receives a Function (as shown with an arrow) from the lower Object (shown with a rectangle). Objects are shown either with ovals (representing the Target Objects) or usually with rectangles (sometimes called the Tool Objects in TRIZ). Attributes of Objects are represented with colors and patterns in the Object symbols. In the middle part of Fig. 2, you would recognize some patterns for solid, powder, liquid, gas, hazardous Attribute, space-varying Attribute, and micro-level Attribute. Nature of Functions is represented by styles of arrows. In Fig. 2, you see arrows standing for a useful Function, a (useful but) insufficient Function, an excessive (hence rather harmful) Function, a harmful Function, and three more Functions with different patterns (for showing their differences). The illustrations in Appendix are designed to stimulate abstract thinking in the 'Objects-Attributes-Functions' scheme. Note that they try to suggest the solution generation scheme of each submethod correctly but do not cover the full range of each submethod.

3.4 How to Use the Table of USIT Solution Generation Methods

The most important usage of the Table of USIT Solution Generation Methods is to study TRIZ in its full aspect of Solution Generation phase. Many TRIZ methods, which are taught in different dedicated chapters in traditional textbooks, can be studied here in a unified way. For the students/practitioners of TRIZ, the fully expanded version [14] containing the descriptions of TRIZ submethods would be most instructive and self-learnable.

For the beginners of USIT, the brief guidelines down to the submethod level (e.g., (1a), (1b), etc.) may be enough. Various simple examples and illustrations should be added for easier understanding. Representative schemes of operations of submethods are shown in simple illustrations in the Appendix for this purpose.

To serious USIT students, the full table shown in the Appendix should be presented and explained. A number of examples should be shown to them with the help of examples already accumulated in TRIZ. Studying TRIZ itself should be encouraged, even though not mandatory for them to understand and apply USIT successfully.

During the USIT sessions for solving real problems, the outline table (i.e., Table 1) may be useful as a brief reference. In the spirit of USIT, we should better use the knowledge in our own brain. Once USIT students learn the full table, they may be able to use the outline table as a reminder without depending on it too much.

The USIT procedure shown in Fig. 1 can be applied without change. Problem solving with USIT must have become much more powerful than before. Applying various submethods to the Objects, Attributes, Functions, and preliminary Solutions would give multiple (rather a large number of) conceptual solutions. It is not necessary of course to make exhaustive trials of the submethod operations onto the elements in the system.

We have been proposing to use USIT and TRIZ software tools in a complementary way [11]. Use USIT as the leading process of the problem solving and as the guidelines for human thinking, especially in a group work in industrial problem solving. Use TRIZ software tools basically as a knowledge-base for supplementing scientific/technological knowledge, good examples of applying principles, and checklists of properties/functions, etc. In our table of USIT Solution Generation Methods, you may find several points where supplements from knowledge-base tools would be effective. We think it not effective to make software tools guide our thinking process and to use software tools as the main tool in a group-work session.

4. DISCUSSION

The full range of TRIZ methods for Solution Generation phase, including Inventive Principles, Inventive Standards, Trends of Evolution, Separation Principle, Trimming, and Self-X principle, have been reorganized in the framework of USIT and unified smoothly into USIT's five Solution Generation Methods. These Solution Generation Methods can be utilized in a compact scheme of USIT process as shown in the flowchart in Fig. 1. The USIT process is easier to learn and apply than ARIZ and many other overall procedures proposed in the framework of TRIZ.

The reorganization and unification of TRIZ methods has made them much simpler to apply to problem solving. Inventive Principles, Inventive Standards, and Trends of Evolution, for instance, can be used simultaneously in an enhanced way.

The prerequisites for using the present USIT Solution Generation Methods may be the understanding of relevant Objects, Attributes, and Functions in the system and of Space and Time characteristics of the system. Such prerequisites should be prepared earlier in the Problem Definition and Problem Analysis phases.

Let us now consider how such prerequisites could be obtained. In case of traditional TRIZ, each Solution Generation method has its own specific Problem Analysis tool: they are Contradiction Matrix for Inventive Principles, S-Field Analysis for Inventive Standards, and ARIZ for Separation Principle. The separation of these pair-wise tools has been making the analysis (i.e. obtaining the prerequisites) in each method insufficient in its scope. Hence, the solution generation becomes difficult and tricky, and the whole process of TRIZ is difficult to learn.

Thus, the unification of TRIZ Solution Generation methods in the present study urges, as the next step, the unification/simplification of TRIZ in the Problem Analysis phase. The present authors understand that USIT has already provided a good proposal for that purpose, too.

As shown in Fig. 1, USIT has clear processes of Problem Definition Stage and Problem Analysis Stage. The Closed World Method urges the problem solvers to find the 'Objects-Attributes-Functions' relationships in the problem system, and the Space and Time Characteristics Analysis makes them aware of the characteristic nature of the problem system in space and in time. Particles Method further makes the images of the ideal solution and how to approach to it. Thus USIT prepares the above mentioned prerequisites of solution generation in a smooth and consistent way.

The present table of USIT Solution Generation Methods has not been shown yet to USIT students. We should be careful to show it in a way as simple and understandable as possible. Once in an early stage of the present work, we made a matrix presentation of the mapping from the Inventive Principles onto the USIT methods. When we showed it to engineers at our USIT practice session, the engineers found it too much for using in practice. This experience gives us a warning against providing too extensive set of tools. Thus we are currently planning to show and use the present information in a step-wise manner as described in Section 3.4.

5. CONCLUSION

The huge body of TRIZ Solution Generation methods, including Inventive Principles, Inventive Standards, Trends of Evolution, Separation Principle, etc., has been reorganized in the present study into simple five Solution Generation Methods in USIT. The latter is shown in the table in Appendix which contains brief and plain guidelines for application. Since the whole procedure of USIT is simple and easy to apply, the present work provides a practical basis for learning and applying the essence of TRIZ in real problem solving in industrial situations.

References[*2]

[1] Altshuller, Genrich S., (1999) 'The Innovation Algorithm', Technical Innovation Center, Worchester, MA, USA, ISBN 0964074044 (E).
[2] Salamatov, Yuri, (1999) 'TRIZ: The Right Solution at The Right Time', Insytec B.V., Hatten, The Netherlands, ISBN 90-804680-1-0 (E); Nikkei BP, Tokyo, (2000) (J).
[3] Ideation International Inc., (1999) 'Tools of Classical TRIZ', Southfield, MI, USA, (E).
[4] Savransky, Semyon D., (2000) 'Engineering of Creativity: Introduction to TRIZ Methodology of Inventive Problem Solving', CRC Press, Boca Raton, FL, USA, ISBN 0-8493-2255-3 (E).
[5] Mann, Darrell, (2002) 'Hands-On Systematic Innovation', CREAX Press, Ieper, Belgium, ISBN 90-77071-02-4 (E).
[6] Domb, Ellen and Bellows, William (2002) 'Opportunity Expansion: Enhancing TRIZ with Dr. Deming's Philosophy', TRIZCON2002: Fourth Annual Altshuller Institute for TRIZ Studies International Conference, Apr. 30 - May 2, St.Louis, MO, USA, (E).
[7] Horowitz, Roni, (2001) 'From TRIZ to ASIT in 4 Steps', TRIZ Journal online, Aug. 2001 (E); TRIZ HP Japan, Sept. 2001 (J).
[8] Horowitz, Roni, (2001) 'ASIT's Five Thinking Tools with Examples', TRIZ Journal online, Sept. 2001 (E).
[9] Sickafus, Ed. N., (1997) 'Unified Structured Inventive Thinking: How to Invent', Ntelleck, Grosse Ile, MI, USA, ISBN 0-9659435-0-X (E).
[10] Nakagawa, Toru, (2001) 'Learning and Applying the Essence of TRIZ with Easier USIT Procedure', ETRIA World Conference: TRIZ Future 2001, Nov. 7-9, Bath, UK, (E); TRIZ HP Japan, Nov. 2001 (E), Aug. 2001 (J).
[11] Nakagawa, Toru, (2002) 'Experiences of Teaching and Applying the Essence of TRIZ with Easier USIT Procedure', TRIZCON2002: Fourth Annual Altshuller Institute for TRIZ Studies International Conference, Apr. 30 - May 2, St.Louis, MO, USA; TRIZ HP Japan, May 2002 (E); Jan. 2002 (J).
[12] Nakagawa, Toru (Ed), 'TRIZ Home Page in Japan' online, [http://www.osaka-gu.ac.jp/php/nakagawa/TRIZ/eTRIZ/] (in English), [http://www.osaka-gu.ac.jp/ php/nakagawa/TRIZ/] (in Japanese). (Note: These are abbreviated here as 'TRIZ HP Japan'.)
[13] Karasik, Y. B., (2000) 'On the History of Separation Principles', TRIZ Journal online, Oct. 2000.
[14] Mann, Darrell (2001) 'Ideality and 'Self-X'', ETRIA World Conference: TRIZ Future 2001, Nov. 7-9, Bath, UK, (E); TRIZ HP Japan, Mar. 2002 (J).

Note *2: (E): written in English, and (J): written in Japanese.

About authors:

Toru Nakagawa: Professor of Informatics at Osaka Gakuin University. Since he was first exposed to TRIZ in May 1997, he endeavored to introduce it into Fujitsu Labs for which he was working. After moving to the University in April 1998, he has been working for introducing TRIZ into Japanese industries and academia. In November 1998 he founded the public WWW site "TRIZ Home Page in Japan" and serves as the Editor. He is currently working to introduce USIT as an easier TRIZ procedure.

He graduated the University of Tokyo in chemistry in 1963, studied at its doctoral course (receiving D. Sc. degree in 1969), became Assistant in Department of Chemistry, the University of Tokyo in 1967; he did research in physical chemistry, particularly experiments and analyses in the field of high-resolution molecular spectroscopy. He joined Fujitsu Limited in 1980 as a researcher in information science at IIAS-SIS and worked for quality improvement of software development. Later he served as a managing staff in IIAS-SIS and then in R&D Planning and Coordination Office in Fujitsu Labs.

E-mail: nakagawa@utc.osaka-gu.ac.jp

Hideaki Kosha: Graduated Keio University Graduate School (M.E.) in applied chemistry and joined Fuji Photo Film Co. in 1981. Did research on magnetic recording materials, and since 1994 promoted VE (Value Engineering) in Production Technology Division. Learned TRIZ in 1997 and USIT in 1999, and started promoting the USIT trial projects in 2000 with Y. Mihara. Has conducted more-than-twenty USIT projects in real industrial applications in the company.

Email: hkosha@ashi.seigi.fujifilm.co.jp

Yuji Mihara: Born in 1946, graduated Hokkaido University Graduate School (M.S.) in chemistry, and joined Fuji Photo Film Co. in 1971. Did research on photo-sensitive materials, worked for product development, and is currently working for intranet management and TRIZ/USIT promotion. Learned TRIZ in 1998 and USIT in 2000 at Nakagawa's 3-day USIT Training Seminar. Since 2000, he has been conducting the training of USIT and promoting TRIZ/USIT application projects in his company. Also serves as Leader at Users Study Group for Innovation Technology, Mitsubishi Research Institute, in 2000 and in 2002.

Email: yuuji_mihara@fujifilm.co.jp