Jerzy CHRZĄSZCZ
INSTITUTE OF COMPUTER SCIENCE, WARSAW UNIVERSITY OF TECHNOLOGY
15/19 Nowowiejska St., 00-665 Warsaw, Poland

TRIZ inventive principles and computer design

Abstract

TRIZ is systematic approach to innovation, being developed since 1950s. This acronym comes from the original Russian name of the methodology, which is usually translated as *Theory of Inventive Problem Solving* and sometimes abbreviated as TIPS. This methodology offers several concepts and tools supporting identification of problems and generation of solutions, which are directed towards required improvements. This directed approach differentiates TRIZ from other inventive methods, that usually rely on a trial-and-error formula. In the beginning, TRIZ addressed mainly mechanical systems. This paper presents short introduction to TRIZ, describing some basic concepts necessary to show inventive principles in appropriate perspective. These concepts are then used for presenting application examples coming from computer design domain.

Keywords: TRIZ, contradictions, inventive principles, Ideal System.

1. Introduction to TRIZ

TRIZ was created and developed by Soviet inventor Genrikh Saulovich Altshuller (1926-1998). He worked in a patent office and the analysis of highly-processed technical knowledge embodied in patents was an important foundation of the whole methodology. Altshuller browsed thousands of patent descriptions and extracted some common characteristics repeated in many solutions from different areas. After distilling these characteristics on an abstract level, reflecting general ideas of the solutions, he obtained relatively small set of rules, proven to be effective in numerous applications. These rules, known as *inventive principles*, represent models of powerful, generic solutions. The fundamental concept of TRIZ approach is to build adequate model of the problem, find the model of appropriate strong solution and adjust the identified generic solution to the specific problem situation. This allows for "bypassing" the barrier blocking direct path from problem to solution and, at the same time, gives access to the "best practices" hidden behind inventive principles. Several books describe many technical [1], [2] and business [3], [4] TRIZ applications, while software and computer design area are addressed in [5], [6], [7].

TRIZ offers consistent terminology and tools for describing technical systems (i.e. systems not existing in nature) as a hierarchy of sub-systems and components, which may be analyzed in different perspectives, e.g. as components of super-systems. Technical systems are built for delivering particular benefits, they require some resources and they usually generate some unwanted effects, such as noise, pollution, etc. Altshuller proposed a simple measure for indicating system maturity, which he called *ideality*. It is calculated as a quotient of sum of benefits related to sum of costs and harms. This formula is mainly used as a conceptual construct – unless all parameters are expressed using same units.

In spite of its qualitative interpretation, the ideality equation is important, because it clearly indicates that system ideality may be affected by any of the parameters involved. It also supports systematic identification of benefits, costs and harms pertinent to a given problem situation. This, in turn, paves the way to another fundamental TRIZ concept, namely the *Ideal System* denoting a system that delivers all required benefits with no costs and no harmful interactions. Imagining and describing such system is important for establishing direction of the solution search.

We will recall the Ideal System concept later on, but first we will present *contradictions*, recognized by Altshuller to be the proper indicators of the inventive situations.

Traditional engineering approach is usually focused on trade-offs between conflicting requirements. This assumption comes – unconsciously, to a great extent – from the formal education and professional experience, which mainly deals with typical problems being solved in typical ways, and typical result of like approach is some improvement achieved by tuning or optimizing the system. TRIZ, on the contrary, aims at satisfying or bypassing conflicting requirements instead of compromising them. This usually requires innovative thinking and the strongest solutions provide great benefits with small changes to the existing system. Moreover, while traditional engineering typically involves adding components or functions to introduce improvement, TRIZ is oriented towards simplification of the system and using easily available resources.

2. Formulating and solving contradictions

At the business level, demand for a new solution is usually articulated as an *administrative contradiction* by indicating a particular need or disadvantage of an existing system, e.g.:
computer generates too much noise,
so that we need to eliminate the noise or to decrease its level.

When it comes to looking for possible solutions, there are typically some conflicting characteristics, such that improving one parameter of the system worsens another important parameter, creating what is called a *technical contradiction,* for instance:
IF we lower processor's clock frequency
THEN the fan noise level will decrease (+),
BUT the performance of the computer will decrease (-).

The most precise manner of describing a conflict of requirements is a *physical contradiction*, which refers to justified contradicting demands regarding different states of the same property of the analyzed system – usually two different values of one physical parameter. To continue with the previous example:
clock frequency must be high (to achieve required performance)
AND clock frequency must be low (to eliminate fan noise).

Solving technical contradictions

In addition to revealing inventive principles, the analysis of the patent files also allowed for developing evidence indicating use of particular principles in different problem situations, described as technical contradictions using generalized parameters. Some of these parameters reflect construction and operation of the system (e.g. *weight of moving object, speed, shape*), while the others address life-cycle of the system (e.g. *ease of manufacture, ease of repair*) or its interaction with the environment (e.g. *loss of energy, object-generated harmful factors*). The outcome of this process was the list of 39 generic *typical parameters* and 39×39 table called *contradiction matrix* or *Altshuller's matrix*. Rows of this matrix denote parameters to be improved, columns denote worsening parameters and the cell at the respective cross-section indicates a few of the 40 inventive principles, which proved to be the most successful in solving technical contradictions of a given type.

In order to use the wisdom accumulated in the matrix, one must perform the following "bypass" procedure:

- *describe the analyzed problem as a technical contradiction (this requires problem-specific parameters to be mapped onto the appropriate typical parameters used in the matrix),*
- *find generic solutions using inventive principles recommended in the respective cell of the contradiction matrix,*
- *adjust generic solutions to a given situation, taking into account specific context of the problem at hand.*

The matrix is not symmetrical, because one dimension reflects advantages and the other – disadvantages. It may also be noted, that the main diagonal of the matrix corresponds to the physical contradictions, where conflicting requirements refer to the same parameter. Initially, some cells remained empty, indicating the lack of statistically justified recommendations for given combinations of parameters. But when computerized databases were created, analysis of patent regularities became much easier. Consequently, the extended version of the matrix published in [2] covers 50 technical parameters instead of original 39, with at least 4 principles recommended for every combination (excluding the diagonal). Among the added parameters are: *compatibility, trainability* and *security*. Moreover, similar approach used for business and management problems resulted in building a fully populated 31×31 matrix [3], indexed by business parameters, such as *R&D cost, production risk, support time, customer feedback, convenience*, etc. In both cases, however, there are still 40 inventive principles.

Solving physical contradictions

Although the formula of physical contradiction may be perceived "unsolvable" and counter-intuitive, it touches the very essence of the problem and thus provides strong support for generating candidate solutions. TRIZ offers simple procedure to organize this activity in a systematic way:

- *try applying separation principles*
  - *conflicting requirements may refer to different places in space (separation in space),*
  - *conflicting requirements may refer to different moments in time (separation in time),*
  - *conflicting requirements may refer to different components of the system or super-system (separation in relation),*
  - *conflicting requirements may refer to different levels of the system organization (separation in system level),*
- *try satisfying conflicting requirements,*
- *try bypassing conflicting requirements.*

For each variant there is a short list of recommended inventive principles, which proved to be the most successful for solving similar problems. And if none of the suggested "best" principles lead to an applicable solution, one should check all remaining inventive principles from the list, looking for inspiration.

3. Contradictions and computer design

To illustrate usability of the physical contradictions, we will present a few examples coming from the computer design field.

Managing processor clock

As described before, the need for high performance conflicts with quiet operation of the computer.

Physical contradiction: clock frequency must be high (to achieve required performance) AND clock frequency must be low (to eliminate fan noise).

Ideal System: processor achieves required performance without power consumption.

Near-Ideal System: processor achieves required performance without power consumption *during breaks in processing*.

Separating conflicting requirements in time: high clock frequency during processing AND low clock frequency during breaks in processing.

Serving peripheral devices

In a computer system with many peripherals several devices may require service while processor may only serve one device at a time. Hence all devices compete against each other for processor's time.

Physical contradiction: processor must serve peripherals (to deliver required system functions) AND processor must not serve peripherals (to promptly respond to new requests).

Ideal System: all peripherals are served without consumption of processor's time.

Near-Ideal System: all peripherals are served with *minimal* consumption of processor's time.

Bypassing conflicting requirements: peripherals are served by DMA controller and/or using interrupts AND processor's time is only used for serving the interrupts.

Non-Volatile RAM memory with built-in battery

NVRAM is a static RAM memory combined with a backup battery providing energy for sustaining data stored in the memory array. To prolong useful life of the NVRAM, the battery drain should be minimized – especially before the first use of the chip.

Physical contradiction: battery must be connected to the memory array (to protect the data) AND battery must not be connected to the memory array (to prolong battery life).

Ideal System: battery connects itself to the memory array when data protection is required.

Near-Ideal System: battery connects itself to the memory array when data protection is required *for the first time*.

Separating conflicting requirements in time: chip is manufactured with battery disconnected from the memory array by internal switch AND when external supply is applied for the first time, the switch connects battery to the memory array permanently.

Multi-drop data transmission

Multi-drop mode uses one transmitter and several receivers with connected inputs. Each data block begins with a header byte indicating logical address of the channel, which should receive the data. In all other channels data coming from the common input are irrelevant and should be discarded.

Physical contradiction: each controller must receive all data (to recognize its own identifier) AND each controller must not receive all data (to reduce transmission overhead).

Ideal System: address bytes and data bytes are only received by the controller being the recipient of the transmission.

Near-Ideal System: data bytes are only received by the controller being the recipient of the transmission AND *address bytes are received by all controllers*.

Separating conflicting requirements in time: each controller is set to receive bytes marked as addresses and check them against its own identifier AND only the controller with particular id changes its reception mode to receive all data bytes.

4. Inventive principles and computer design

Inventive principles, just like typical parameters mentioned before, have been extracted from evidence rather than designed as a conceptual construct. Therefore they differ in nature and level of detail. Some of them refer to system structure (e.g. *segmentation, asymmetry, intermediary*) and others address system operation (e.g. *periodic action, feedback, self-service*) or even physical phenomena (e.g. *phase transitions, thermal expansion, color changes, etc.*). Sometimes they may be used directly, but their primary purpose is to inspire creativity by indicating a generic meta-solution. For instance *nested doll* principle stands for any construction with components inserted or hidden into other similar components, what may be related e.g. to layers in a stack of communication protocols.

The following sections describe examples of using selected inventive principles in computer design area.

Segmentation + Universality

System: add-on board for displaying Power-On Self-Test codes generated by BIOS program during PC start-up. System consists of address and bus cycle decoder, data register mapped in I/O space, 2-digit LED display and 2 transcoders for translating nibble hex data into 7-segment code.

Problem: number of I/O signals necessary to provide address, data, strobes and display control required use of 3 PLD chips, which appeared too costly for intended production.

Solution: the decoder was decomposed into 2 half-decoders (segmentation), cross-coupled to enable each other when respective conditions are met [8]. This redesign allowed for not only the fitting of the whole logic in 2 PLDs instead of 3, but also these PLDs had identical contents (universality).

Harm into benefit + Prior action

System: PC diagnostic add-on board for testing interrupt requests and DMA requests on ISA bus. The requests are asserted by changing line state from low to high.

Problem: the PLD chips intended for production required 2 logical cells for creating the flip-flop necessary to monitor single request line, totaling 4 lines per chip.

Solution: using a capacitor in series with a signal input together with a pull-down resistor resulted in a signal differentiator which converted rising edge of the request signal into a positive pulse [9]. Serial capacitors are usually unwanted (if not harmful) in digital circuits, but in this configuration they allowed to double the performance of PLDs, with one chip being able to serve 8 request lines.

Self-service

System: programmable tester of digital modules with several i8255 parallel interfaces controlling and/or sensing external signals. Each controller serves 3 ports that may be configured as inputs or outputs and they are separated from the device under test by bidirectional tristate buffers.

Problem: for proper operation the direction of each buffer should be switched in accordance with the operating mode of the particular port of the programmable controller and to achieve this, additional registers are required for controlling the buffers programmatically. Moreover, improper combination of controls may result in opening of 2 sets of outputs against each other.

Solution: to avoid the hazard of electrical conflict caused by program errors, the buffers are controlled by a PLD that constantly monitors configuration changes applied to a particular controller by software and adjusts the buffer directions on the fly [10]. In this way operation of the system may be both safer and faster.

Feedback

System: battery-powered data logging device with 8051 series microcontroller featuring idle and power-down modes for reducing energy consumption.

Problem: exiting from idle mode requires an interrupt, while power-down– with over 100 times lower energy consumption – may be terminated by a system reset solely.

Solution: external device send sinterrupt signal to reset input via logical gate, which is automatically blocked by a port line changing its state upon reset [11]. Hence, in spite of a pending interrupt request, the reset signal is negated and the system may restart to handle the device, so that simple feedback allows for exiting from power-down mode by interrupt.

Another dimension

System: diagnostic interface for monitoring half-duplex communication in an industrial RS232 data link.

Problem: it is not possible to receive data from both directions using separate UART controllers, because only one COM port is available in the PC approved for on-site monitoring purposes.

Solution: combined signals transmitted from both devices are routed to receiver input and one of the transmission lines is additionally connected to another line of COM port, equipped with an edge detector integrated in UART [12]. Such arrangement allows the diagnostic program to distinguish the sender of a particular data byte using information extracted from the transmission signals in another channel of the same interface.

Dynamization

System: PC diagnostic add-on board for testing interrupt requests (11 lines) and DMA requests (7 lines) on ISA bus. Each of the 18 outputs should be able to assert low and high states as well as remain in high-impedance state not to interfere with other devices connected to system bus.

Problem: implementation of the necessary logic required 4 PLD chips, which appeared too costly for intended production.

Solution: instead of providing a separate control unit for each request line, the logic was redesigned to use only one control unit and switches routing its output to appropriate lines under software control [13]. Change of the operation principle allowed for decreasing the chip count and the total cost of the board.

The other way around

System: control unit of a local-area paging system connecting to standard telephone lines operated by Central Office or Private Branch Exchange (PBX). In order to achieve compliance with telephony regulations, integrated line interface modules were used.

Problem: the customer requested one of the telephone interfaces to accommodate local telephone set, i.e. instead of telephone-like operation, the interface should provide PBX-like operation.

Solution: one of the line modules was reconnected in such a way, that battery voltage appeared on the socket like in PBX line, so that the controller was able to recognize "handset off-hook" condition of the connected telephone set and handle it properly [14].

5. Summary

TRIZ offers several tools that are built upon statistically justified evidence of successful solutions. Using data mining language, one could describe Altshuller's research as *pattern recognition in the corpus of ideas documented in patent files*.

The concept that someone already solved a similar problem and reusing proven solutions is more beneficial then re-inventing them is explored in several TRIZ techniques, making this methodology a powerful booster for systematic innovation in numerous areas, including computer design.

6. References

[1] Gadd K.: TRIZ for Engineers: Enabling Inventive Problem Solving, Wiley 2011, ISBN: 978-0-470-74188-7.

[2] Mann D.: Matrix 2010 – Re-updating TRIZ Contradiction Matrix, IFR Press 2012, ISBN: 978-1-906769-19-2.

[3] Mann D.: Hands-On Systematic Innovation for Business and Management, IFR Press 2014, ISBN: 1-898546-73-8.

[4] Yeoh T. S.: TRIZ - Systematic Innovation in Business & Management, First Fruits Sdn Bhd 2014, ISBN: 978-983-804-035-8.

[5] Mann D. Systematic (Software) Innovation, IFR Press 2008, ISBN: 978-1-906769-01-7.

[6] Rubin M.S., Kiyaev V.I.: Fundamentals of TRIZ and innovation. The application of TRIZ in software and information systems. Textbook. St. Petersburg, 2011. In Russian.

[7] Abramov O.Y.: Trends of Engineering System Evolution in transmission systems and information processing. Proc. TRIZ Developers Summit 2008. St. Petersburg, 2008. In Russian.

[8] Chrząszcz J.: Monitor follows PC's power-on self-test (Design Idea 1295). EDN, September 16,1993, p. 124.

[9] Chrząszcz J.: Capacitor doubles PLD's performance (Design Idea 1752). EDN, September 14, 1995, p. 141.

[10] Chrząszcz J.: Autopilot directs 8255 external buffers (Design Idea 1997). EDN, March 03, 1997, p. 100.

[11] Chrząszcz J.: Use 8051's power-down mode to the fullest (Design Idea 2557). EDN, July 06, 2000, p. 138.

[12] Chrząszcz J.: PC monitors two-way RS-232 transmission (Design Idea 2661). EDN, February 01, 2001, p. 126.

[13] Chrząszcz J.: Analog multiplexers help test ISA bus IRQs and DRQs (Design Idea 1796). EDN, December 07,1995, p. 121.

[14] Chrząszcz J.: DAA circuit emulates central-office operation (Design Idea 2376). EDN,July 08, 1998, p. 98.

Received: 10.12.2017 *Paper reviewed* *Accepted: 01.02.2018*

Jerzy CHRZĄSZCZ, PhD

Assistant Professor in Institute of Computer Science, Warsaw University of Technology. Interested in Computer Graphics, Microprocessor Systems, Programmable Logic, Information Security and Systematic Innovation. Author / coauthor of 7 patents and over 60 publications. Supervisor of over 50 MSc / BSc theses. Certified QMS Lead Auditor (ISO 9001), ISMS Lead Auditor (ISO 27001), BCMSLead Auditor (ISO 22301) and TRIZ Practitioner (MATRIZ Level 2).

e-mail: jch@ii.pw.edu.pl