Application of a Contradiction Table to Computer Architecture
ー Architecture and Processing Technique ー

2007/8/30

T. Shonai Central Research Laboratory, Hitachi Ltd., Japan
S. Kawabe Department of Information Science, Meisei University, Japan*
N. Hamanaka Microdevices Division, Hitachi, Ltd., Japan*

*Note: This work was done when they were engaged at Central Research Laboratory, Hitachi Ltd.

Contents

1. Introduction
   - ･Problems and our approach
   - ･Related works
2. TRIZ word translation for computers; and submatrix
   - ･Case studies: processor, cache, phase adjustment
3. Applications to wider area
   - ･Parallel computers, storage systems, database systems and IT system management
4. Application to real patent generation
   - ･Datacenter architecture (blade systems)
5. Final remarks

1.Introduction 1.1 Problems & our approach

- Many computer/software engineers and researchers say
  - “Is TRIZ effective to our area ?”
- Reason:
  - Contradiction table and invention principles appear to focus on mechanics, structure, physics, and chemistry.
- Our approach
  - Perform case studies of computer-patent generations
  - *Paraphrase* words in improving/worsening parameters and invention principles with words in computer areas
  - *Reduce* a large contradiction table to a smaller one

1.2 Related works

- “TRIZ and Software ー 40 Principle Analogies” [Rea]
  - Not easy to use because of the difference in fields from ours
- “TRIZ Promoting Activities in NEC” [Suzuki, et al.]
  - Focusing on organizational activities
- “Re-structuring TRIZ to Software Engineering” [Mann]
- “Software Engineering and TRIZ” [Nakagawa]
  - Focusing on different fields of software-development processes
- “Matrix 2003”[Mann, et al.]
  - New improving/worsening parameters for computers and other recent fields.
  - Inventive principles are, however not changed.

★Computer and software fields are huge.
★Previous works are almost all “software engineering”
★Others’ works may not be applicative to your/my application fields.
★Tailoring should be done by our/yourselves.

2. Case studies (analysis after inspiration) (1) processor

- Purpose
  - Reduction of bank conflicts in multi-banked caches occurred by pairs of load instructions.
- TRIZ application
  - Worsening parameter → [6] increase of area（because chip area is increased by multi-ported caches）
  - Improving parameters → [25] loss of time（performance increase）
  - Inventive principles → [10] preliminary actions, [35] state transformations, [17] another dimension, [4] asymmetry
- Main idea
  - By using a conflict-history table, it is possible to predict and issue a pair of load instructions that have made no bank conflict in previous cache executions.
- Analysis from TRIZ view-point
  - Act preliminarily [10] by predicting state transformation[35] !
- Interpretation and comments
  - This kind of method—act preliminarily by predicting state transformation — has been frequently used in making faster processors/computers since 1960s. For example, branch prediction is used in almost all modern processors. Other kinds of prediction techniques (like load-value prediction) have been researched.

(2) Fast-cache-access technique

- Purpose
  - In a symmetric multi-processors system, reduction of store throughput is a problem because lots of store accesses issued by many processors have to be written to each cache. Store-in-cache is therefore necessary. Since a single shared large cache is unpractical because of implementation problems, it has to be divided into many smaller store-in-caches, and the cache-coherent-protocol overhead has to be reduced.
- TRIZ application
  - Worsening parameter → [24] loss of information（failure of cache coherent protocol）
  - Improving parameters → [25] loss of time（performance increase）
  - Inventive principles → [24] intermediary、[26] copying、[28] mechanics substitution、[32] color changes
- Main idea
  - Using prediction of data-access patterns with access-pattern history can optimize the cache-coherency behavior dynamically.
- Analysis from TRIZ view-point
  - Coloring data [32] —by using prediction of the data-access pattern with access-pattern history—and copying[26] and sharing the colored data.
- Interpretation and comments
  - “color change”[32] is often used and can be interpreted in computer architecture/logic to mark data with tags and distinguish the data from others. It is an essential technique and has been used very frequently.

(3) Cache organization

HITACHI Inspire the Next

- Purpose
  - Because the capacity of caches is limited by chip sizes, it is important how to improve the cache hit ratio with cache of limited size. It may be effective to find out data with high locality and treat them in some special manner.
- TRIZ application
  - Worsening parameter → [24] loss of information（failure of cache coherent protocol）
  - Improving parameters → [25] loss of time（performance increase）
  - Inventive principles → [24] intermediary、[26] copying、[28] mechanics substitution、[32] color changes
- Main idea
  - To divide all the data into a few classes of data, and store some classes of data in a cache prior to the remaining classes of data. The priority can be set and changed according to the characteristics of a system and programs in each partition.
- Analysis from TRIZ view-point
  - Coloring data [32] ―by distinguishing data with high locality in order to change the priority at an intermediate point[24](i.e., a switching point) partition by partition at the time of system generation.
- Interpretation and comments
  - The inventive principle “intermediary [24]” is also often used. It can be interpreted as the time of block transfer, an interruption, or other events involved. Considering a switching point may inspire new ideas.

(4)Phase-adaptation circuit

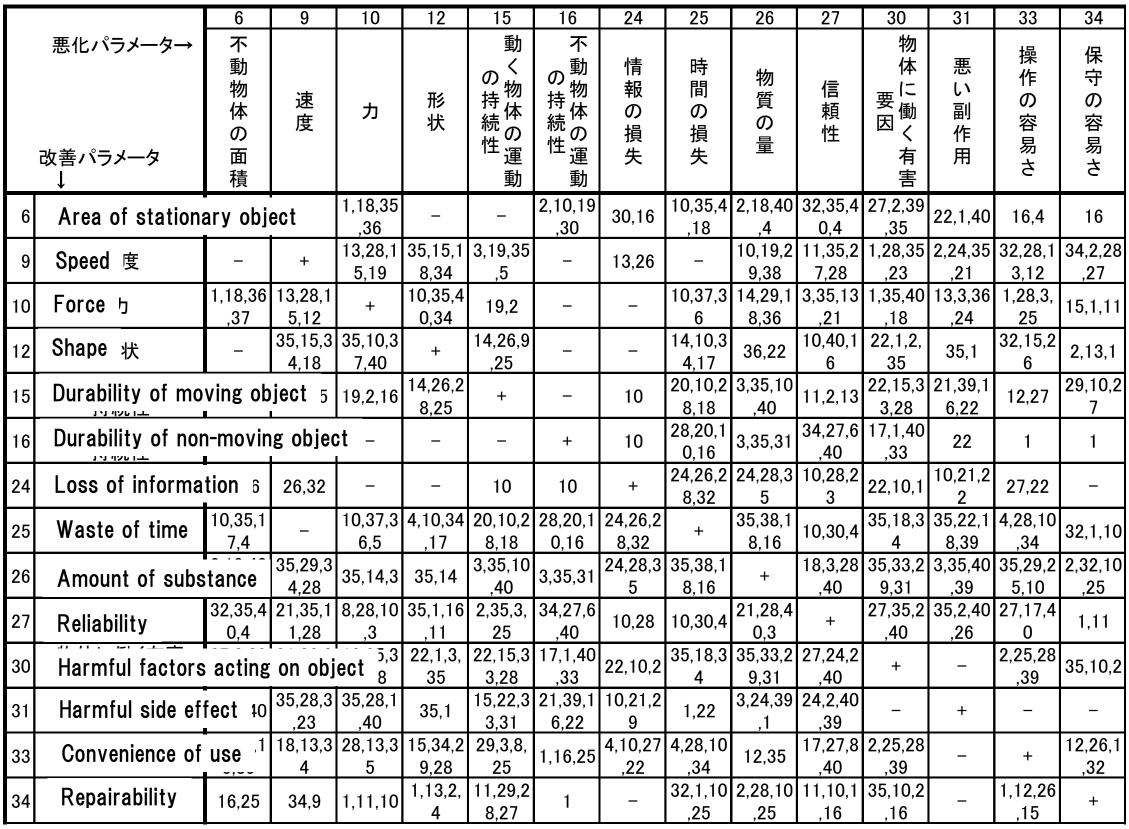
HITACHI Inspire the Next

- Purpose
  - To reduces the skew variation of signal interface circuits between LSIs. The skew variation at the chassis interface of multiple bits with long propagation time is very large.
- TRIZ application
  - Worsening parameter →[9] increase of chip area
  - Improving parameters →[31] other harmful side-effect（ skew variation becomes large ）
  - Inventive principles →[22] “blessing in disguise” [1] segmentation, [40] composite material
- Main idea
  - To detect and memorize phase shifts among data signals and parallel distributed clocking signals at plural detection points, and to compensate the phase shift of a data signal.
- Analysis from TRIZ view-point
  - [22] “blessing (phase shifts) in disguise (phase compensation)” [1] segmentation（control by detection at plural points）
- Interpretation and comments
  - This association is time-consuming. Although [31] “other harmful side-effects” was too abstract to associate something, we finally associated it with increase of skew variation. [22] “blessing in disguise” can be easily associated with the main idea in analysis after inspiration. However, it is difficult to get the main idea from scratch only with “blessing in disguise”. Doing so needs widespread and deep technical knowledge.

2.2 Summary of case studies

- Paraphrase of inventive principles
  - [32] color changes → Marking data with tags and distinguishing the data from others.
  - [24] intermediary → Interpreted as the time for the events involved, such as block transfers and interruptions. Considering a switching time point may inspire new ideas.
- Paraphrase of improving/worsening parameters
  - More performance → [25] loss of time、
  - Increase number of logic gates or chip area → [6] area of stationary object
  - Cache coherency → [24] loss of information
- Submatrix
  - There are some parameters with less relation to architecture and processing techniques. These parameters are deleted and a submatrix with only 14 parameter's is made.
  - The area of the submatrix is a quarter of the original one.
  - The submatrix is so small that it can be included in an A4 sheet or on a PC display with 800x600dots, and it is easily used by beginners using TRIZ.

(Con't) Submatrix



悪化パラメータ→ 改善パラメータ ↓		6	9	10	12	15	16	24	25	26	27	30	31	33	34
		不動物体の面積	速度	力	形状	動く物体の運動の持続性	不動物体の運動の持続性	情報の損失	時間の損失	物質の量	信頼性	物体に働く有害要因	悪い副作用	操作の容易さ	保守の容易さ
6	Area of stationary object			1,18,35,36	–	–	2,10,19,30	30,16	10,35,4,18	2,18,40,4	32,35,40,4	27,2,39,35	22,1,40	16,4	16
9	Speed 度	–	+	13,28,15,19	35,15,18,34	3,19,35,5	–	13,26	–	10,19,29,38	11,35,27,28	1,28,35,23	2,24,35,21	32,28,13,12	34,2,28,27
10	Force ｶ	1,18,36,37	13,28,15,12	+	10,35,40,34	19,2	–	–	10,37,36	14,29,18,36	3,35,13,21	1,35,40,18	13,3,36,24	1,28,3,25	15,1,11
12	Shape 状	–	35,15,34,18	35,10,37,40	+	14,26,9,25	–	–	14,10,34,17	36,22	10,40,16	22,1,2,35	35,1	32,15,26	2,13,1
15	Durability of moving object		5	19,2,16	14,26,28,25	+	–	10	20,10,28,18	3,35,10,40	11,2,13	22,15,33,28	21,39,16,22	12,27	29,10,27
16	Durability of non-moving object			–	–	–	+	10	28,20,10,16	3,35,31	34,27,6,40	17,1,40,33	22	1	1
24	Loss of information	6	26,32	–	–	10	10	+	24,26,28,32	24,28,35	10,28,23	22,10,1	10,21,22	27,22	–
25	Waste of time	10,35,17,4	–	10,37,36,5	4,10,34,17	20,10,28,18	28,20,10,16	24,26,28,32	+	35,38,18,16	10,30,4	35,18,34	35,22,18,39	4,28,10,34	32,1,10
26	Amount of substance		35,29,34,28	35,14,3	35,14	3,35,10,40	3,35,31	24,28,35	35,38,18,16	+	18,3,28,40	35,33,29,31	3,35,40,39	35,29,25,10	2,32,10,25
27	Reliability	32,35,40,4	21,35,11,28	8,28,10,3	35,1,16,11	2,35,3,25	34,27,6,40	10,28	10,30,4	21,28,40,3	+	27,35,2,40	35,2,40,26	27,17,40	1,11
30	Harmful factors acting on object			5,3 8	22,1,3,35	22,15,33,28	17,1,40,33	22,10,2	35,18,34	35,33,29,31	27,24,2,40	+	–	2,25,28,39	35,10,2
31	Harmful side effect	40	35,28,3,23	35,28,1,40	35,1	15,22,33,31	21,39,16,22	10,21,29	1,22	3,24,39,1	24,2,40,39	–	+	–	–
33	Convenience of use	,1	18,13,34	28,13,35	15,34,29,28	29,3,8,25	1,16,25	4,10,27,22	4,28,10,34	12,35	17,27,8,40	2,25,28,39	–	+	12,26,1,32
34	Repairability	16,25	34,9	1,11,10	1,13,2,4	11,29,28,27	1	–	32,1,10,25	2,28,10,25	11,10,1,16	35,10,2,16	–	1,12,26,15	+

3. Applications to wider areas

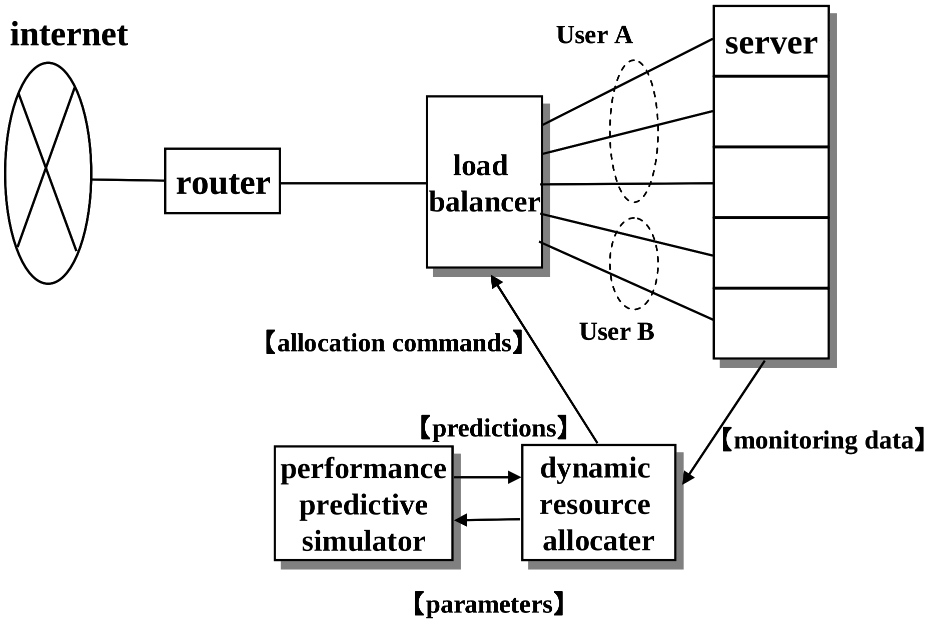
- Wider application areas:
  - Parallel computers, storage systems (RAID, NAS), database systems, IT system management, and image/voice/text processing.
- New paraphrases and application examples are added.
- Update of submatrix
  - Four parameters are added: [32] manufacturability，[36] complexity of device，[37] complexity of control，[39] productivity.
- Significance of paraphrase of inventive principles
  - Essence of technical knowledge of each field
  - Researchers with considerable experience usually accumulate problem-solving strategies as implicit knowledge/wisdom that is exploited frequently in technical problem-solving in their field/organization.
  - Paraphrase of inventive principles can formulate this implicit knowledge/wisdom and can be used to teach it to subordinates or newcomers to a technical field.

4. Application to real patent generation(1/4)

- Problems at internet datacenters (iDCs)
- Response degradation due to surges in internet traffic became a problem
  - Frequent shortage of “i-mode” services due to excess user usage became a social problem(1999-2000)
- We innovated new ideas.
  - Virtual Private Data Center (VPDC)
  - A datacenter architecture that can share resources such as servers, storage, and network bandwidth with different users and applications in a cost-effective manner and can respond to surged internet accesses in excess of the prior expected amount of accesses by using dynamic resource allocation.
- We used TRIZ in the process of making patents from the above basic idea.

4. Application to real patent generation(2/4)

- **Problem formulation**
  - ① When internet accesses surge, servers must be reallocated and must supply optimal throughput.
  - ② Number of servers must be decreased so as to be cost-effective.
  - ③ Optimal reallocation must be done without lots of human operations.



4. Application to real patent generation(3/4)

- Contradiction problem（１）
  - Improving: to increase throughput by optimally reallocating resource like servers‥‥①
  - Worsening: to decrease the number of servers‥‥‥‥‥‥‥②
- Inventive principles
  - [35] state transformation 、[38] strong oxidants、[18] mechanical vibration、[16] partial/excessive actions
- Interpretation
  - TRIZ advises the inventor to act partially/excessively by state transformation (prediction) while considering periodic or wavy actions.
  - “State transformation” may be efficiency change of CPU, memory, and IO of servers, which should be monitored.
  - “Partial/excessive actions” are thought to be optimal allocations of a limited amount of resources, e.g., servers.
  - “Periodic or wavy actions” may be periodic variation of loads.
- Optimal allocation and ③ become another contradiction.

4. Application to real patent generation(4/4)

- Contradiction problem（2）
  - Improving: to reallocate optimally with fewer or no human operations・・・・・③
  - Worsening: optimal reallocation requires more time.
- Inventive principles
  - [4] asymmetry、[28] mechanical substitution、[10] preliminary action、[34] discarding and recovering
- Interpretation
  - To reallocate optimally in advance by prediction with modeling and simulation, or actual measurements (preliminary action)
  - To reallocate optimally by discarding/reusing servers (discarding and recovering)

- Parts of this idea are used in *Hitachi's Service Platform Concept Harmonious Computing* and *Integrated Service Platform BladeSymphony*.

5. Final remarks

- Conclusion
  - I have described paraphrasing of words in inventive principles and submatrixing when contradiction table and inventive principles are applied to computer problems.
  - These methods are effective when we use TRIZ in our department at Hitachi.
- However
  - They do not guarantee good ideas.
  - Application of TRIZ with superficial understanding of technical problems and ideas results in no gain.
  - It is necessary to understand technical backgrounds and whole problems deeply, choose appropriate TRIZ tools depending on situations, and abandon previous ways of thinking.
- About Matrix 2003
  - Its improving/worsening parameters reflect recent US computer patents.
  - Although the inventive principles are the same as the older ones, the approach described here is thought to be effective with Matrix 2003.