…のダイジェスト

ICALP Masterclass Talk: Algorithms and Complexity for Japanese Puzzles

Ryuhei Uehara

Japan Advanced Institute of Science and Technology

uehara@jaist.ac.jp

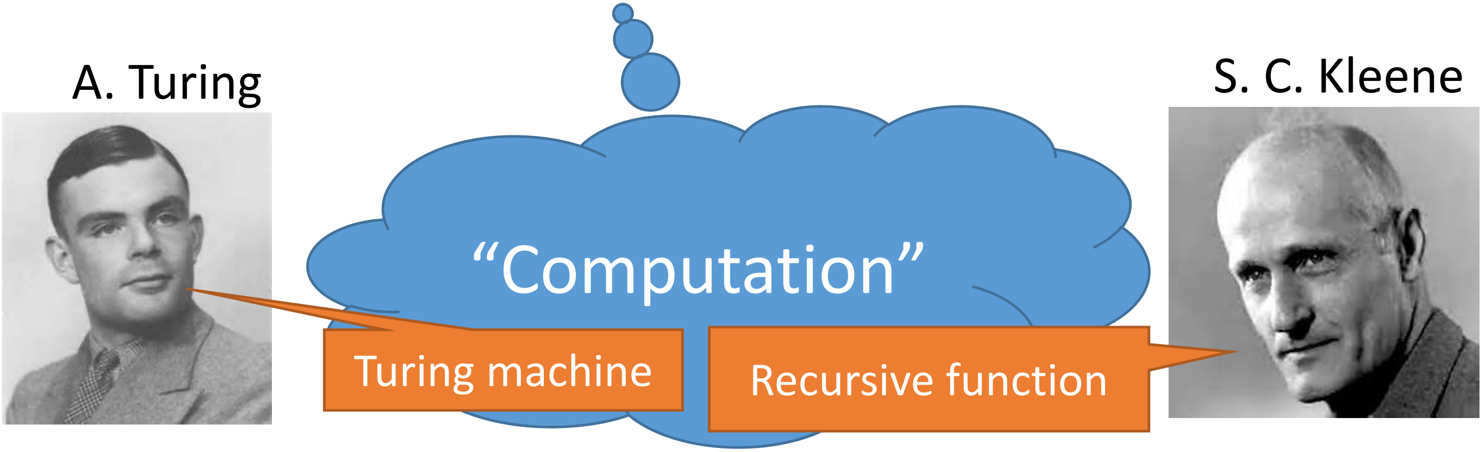
http://www.jaist.ac.jp/~uehara

Complexity v.s. Puzzles & Games

1. Computational Complexity v.s. Puzzles & Games
2. Complexity Classes characterized by Games and Puzzles
   - Classic
     - Historical results
   - Modern
     - What have been considered?
   - Recent and Future
     - What problems on the edge?

Computational Complexity v.s. Puzzles & Games

- What's "computation" could be… (1930s-1940s)



To consider "computation," what we need is

- Basic operations (=model of computation)
- How can we combine them (=algorithms)

Computational Complexity v.s. Puzzles & Games

- What's "computation" could be… (1970s)

"Computation"

Games and Puzzles Can Be!!

John Horton Conway

To consider "computation," what we need is

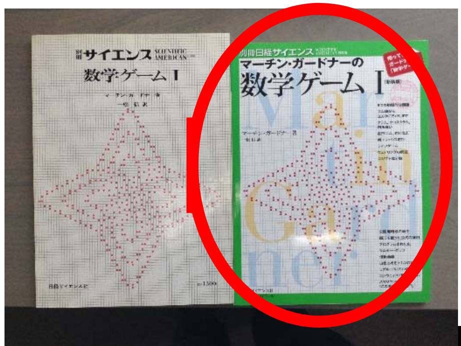
- Basic operations (=model of computation)
- How can we combine them (=algorithms)

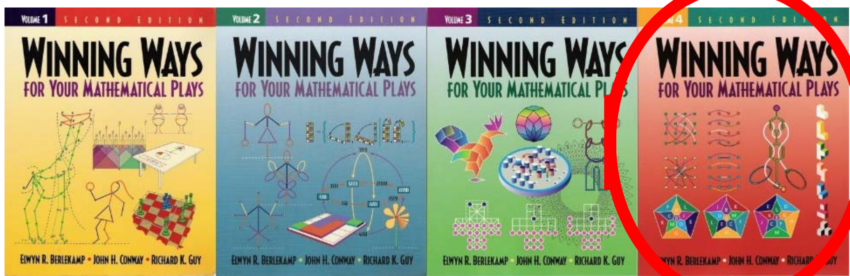
Computational Complexity v.s. Puzzles & Games

- What's "computation" could be… (1970s

  Conway's *Game of Life* (1970)

  - For young guys
    - It is a kind of cellular automaton with quite simple rules.
    - It is "*Universal*", that is, it computes any function!
      - Some nice books:





Simon J. Fraser, John "Horned" (Horton) Conway, *1975*

Computational Complexity v.s. Puzzles & Games

- What's "computation" could be… (1970s

  Conway's *Game of Life* (1970)

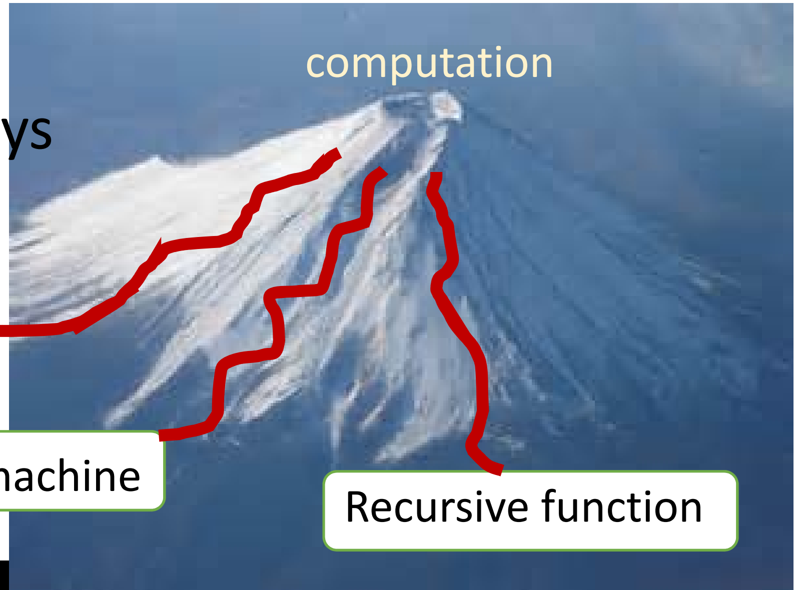
  - For young guys
    - It is a kind of cellular automaton
      with quite simple rules.
    - It is "*Universal*",
      that is, it computes any function!
  - For veteran folks
    - Quite fancy software "Golly"

Simon J. Fraser, John "Horned" (Horton) Conway, *1975*

Computational Complexity v.s. Puzzles & Games

- Puzzles and Games to consider “computation”
  - Simple and Uniform (with reasonable model)
  - That may extract the <u>essence</u> of the difficulty of some computation
  - That may give us <u>new aspect</u> of some computation

⇒Same goal through different ways



Short Ads.

- In JAIST, we have “JAIST Gallery” that has around 10000 puzzles called *NOB’s Puzzle Collection*!

I’m a director of this gallery!

NOB Yoshigahara
(1936-2004)

Classic Results (1970s～1980s):

- Game to consider "computation"
  - Characterization by artificial game
  - *Pebble game* (though we have many variants)

    Input: Directed graph *G,* placement of "*pebbles*"

    Rule: Move pebbles along edges and remove some pebbles in certain rules

    Output: Determine if you can move a pebble to a goal
  - It is complete for some computational classes;
    - NLOG, P, NP, PSPACE, EXP

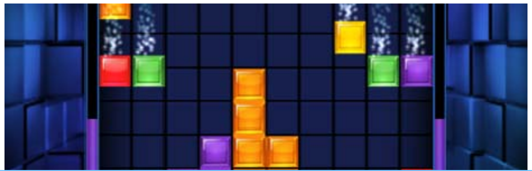
1 player/2 players
Number of pebbles
Acyclic or not

- References:

  - J. Hopcroft, W. Paul and L. Valiant. "On Time versus space," *J. Assoc. Comput. Mach*. 1977
  - Richard J. Lipton and Robert E. Tarjan. "Applications of a Planar Separator Theorem," *SIAM J. Comput*. 1980
  - Stephen Cook; Ravi Sethi. "Storage requirements for deterministic polynomial time recognizable languages". *Journal of Computer and System Sciences,* 1976.
  - Takumi Kasai; Akeo Adachi; Shigeki Iwata. "Classes of pebble games and complete problems". *SIAM Journal on Computing,* 1979.

More Classic Results (1980s～):

- Puzzles to consider “computation”
  - Characterization by *natural* games and puzzles
  - *Many puzzles and games*
    - *E.g.,* Geometry (しりとり), Solitaire, Crossword puzzle, Jigsaw puzzle (matching puzzle), UNO, Video games, Pencil puzzles, …



So far ... (1980s～2000s):

- We had tons of *X*-Complete problems;
  - NP-complete puzzles

    1 player, something decreases in each step
    - Tons of papers…
  - PSPACE-complete / EXP-complete games

    2 player version of these NP-complete problems

Interesting Puzzle ≒ NP-C

- They give some insight of these classes

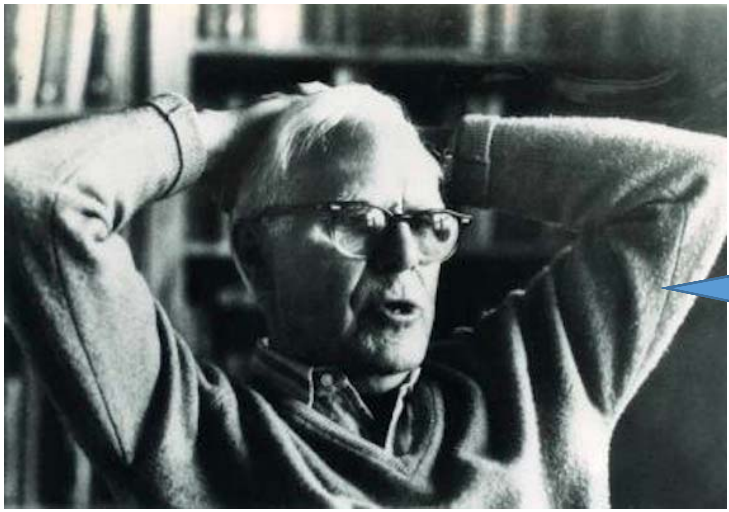
  NP: 1 player, something decreases in each step.

  PSPACE: 2 players (…alternating Turing Machine)

- We needed some general model for them…

So far … (1980s～2000s):

- Still unsolved
  - Sliding Block puzzles like “Daddy Puzzle”, “Sokoban”
- Martin Gardner said that…

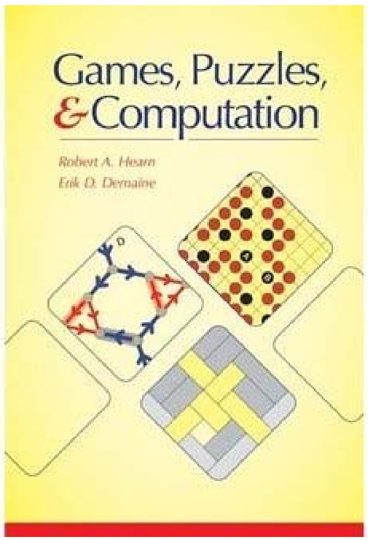


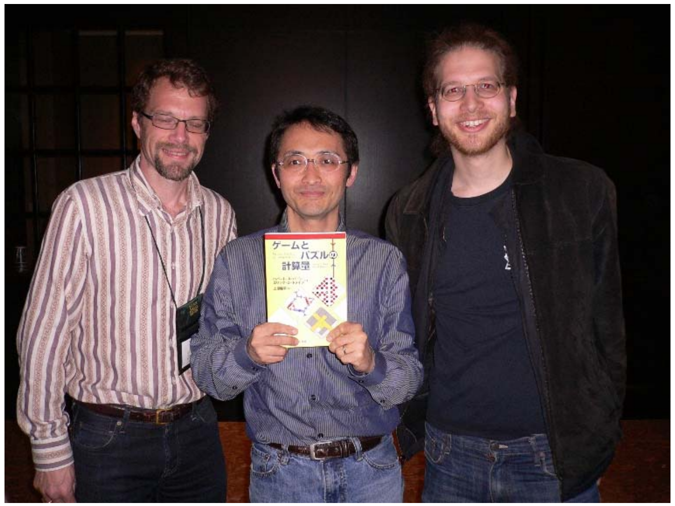
“These puzzles are very much in want of a theory”
Scientific American 210 (1964)

… 40 years later,

Modern Results (2010s～):

- New framework to consider “computation”
  - “Constraint Logic” by <u>Bob Hearn</u> and Erik D. Demaine
  - *Essentially, game (2player) and puzzle (1player)*
    - *That can model many previous known games and puzzles,*
    - *And solves the open problems about sliding block type puzzles.*





Modern Results (2010s～):

- New framework to consider "computation"
  - "Constraint Logic" by <u>Bob Hearn</u> and Erik D. Demaine
  - *Roughly, it is a game on a graph*

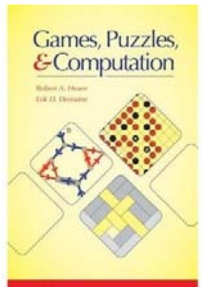
    Input: Directed graph *G*, each edge has weight and direction

    Rule: Each vertex is *balanced*, an operation is flipping an edge

    Output: Determine if you can flip some specified edge
  - Relatively higher classes:

#flips of an edge

	0 player	1 player	2 player	Team, imperfect information
Unbounded	PSPACE	PSPACE	EXPTIME	RE (undecidable)
Bounded	P	NP	PSPACE	NEXPTIME

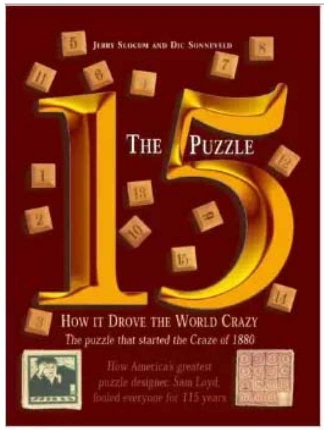


Some remarkable puzzles…

- Finally solved
  - Sliding Block puzzles are PSPACE-complete.
  - Unlike other NP-complete problems, it can recover the same state many times… that property makes them to be PSPACE-complete?
- It reminds us a classic puzzle solved in 1990,,,
  - 15 puzzle
  - It has a long and funny stories;

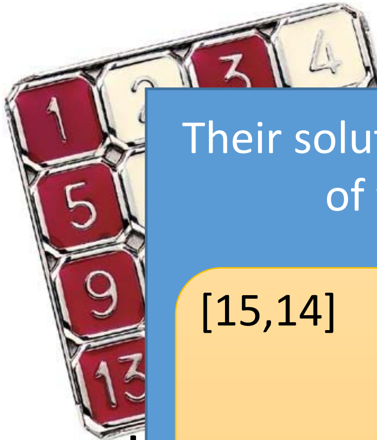
    see “The 15 Puzzle Book”

    by Jerry Slocum, 2006.



Top puzzle collector in the world…

Some remarkable puzzles…



- The 15 Puzzle

  It is easy to generalized to n × n board

  Input: Two arrangements ***s*** and ***t*** of the nu

  Goal: Slide a panel from ***s*** to ***t***

  Output: …

  Yes/No: Linear time by parity check

  If Yes, find a sequence of arrangements: $O(n^2)$ time

  Furthermore, find a sequence: $O(n^3)$ time

  However, find a shortest sequence: NP-complete!!

Their solution space consists of two groups

[15,14]

1	2	3	4
5	6	7	8
9	10	11	12
13	15	14	

[14,15]

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

- Reference:
  - Daniel Ratner and Manfred Warmuth. "The (n^2-1)-Puzzle and Related Relocation Problems," *J. of Symbolic Computation*, 1990.

Recent and Future Results (2010s～):

- *New concept* of problems to consider "complexity" inspired by these puzzles:

Reconfiguration Problems

Input: Problem P, two feasible solutions S_1 and S_2

Operation: Simple rule for modification of a solution

Decision	Problem 1: Determine if S_1 can be transformed to S_2
Find	Problem 2: Find a sequence of solutions between S_1 and S_2
Shortest	Problem 3: Find a shortest sequence between S_1 and S_2

Recent and Future Results (2010s～):

- *New concept* of problems to consider "complexity" inspired by these games/puzzles:

Reconfiguration Problems

	Sliding Block puzzle	$(n^2$-1) puzzle
Decision	PSPACE-complete	Linear
Find	(PSPACE-complete)	Poly-time
Shortest	(PSPACE-complete)	NP-complete

Many problems should be here
And they give some new insight of the classes

Recent and Future Results (2010s～):

- Not game-like results for reconfiguration problems:
  - SAT: "Decision problem" is PSPACE-complete

    Reference:

    P. Gopalan, P.G. Kolaitis, E.N. Maneva, C.H. Papadimitriou, "The connectivity of Boolean satisfiability: computational and structural dichotomies," *SIAM J. Comput*. 2009.
  - IS, Clique, Vertex Cover, Set Cover, IP: "Decision problem" is PSPACE-complete

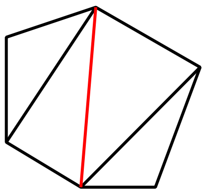
    Reference:

    T. Ito, E. D. Demaine, N. J. A. Harvey, C. H. Papadimitriou, M. Sideri, R. Uehara, and Y. Uno: On the Complexity of Reconfiguration Problems, *Theoretical Computer Science*, 2010.

In my measure, "Sliding-block puzzle type"

Recent and Future Results (2010s～):

- Bit game-like result for reconfiguration problems:
  - Famous open problem in Computational Geometry

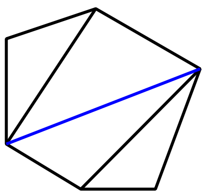


Input: Simple polygon, two triangulations T_1, T_2

Operation: “flip” one diagonal

Known: Every T_1 is flippable to T_2 in $O(n^2)$ flippings

Question: Find a shortest flipping



Result: It is NP-complete!!

It was open 40 years like sliding block puzzle…

Reference:

O. Aichholzer, W. Mulzer, A. Pilz, Flip distance between triangulations of a simple polygon is NP-complete, ESA 2013.

In my measure, “(n^2-1) puzzle type”

Recent and Future Results (2010s～):

- Not game-like, but something remarkable:
  - SAT: Trichotomy for the classes P, NP, and PSPACE from the viewpoint of "Shortest problem"

    Reference:

    A. E. Mouawad, N. Nishimura, V. Pathak and V. Raman: Shortest Reconfiguration Paths in the Solution Space of Boolean Formulas, *ICALP 2015*, 2015/7/8.

In my measure, this one may be the first example between "Sliding-block puzzle type" and "(n^2-1) puzzle type".

Recent and Future Results (2010s～):

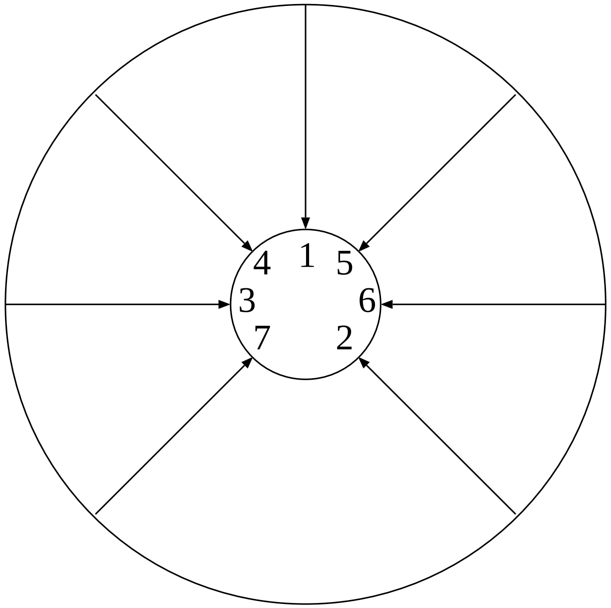
- More game-like, but not settled problem:
  - Amida-kuji (Ladder Lottery)

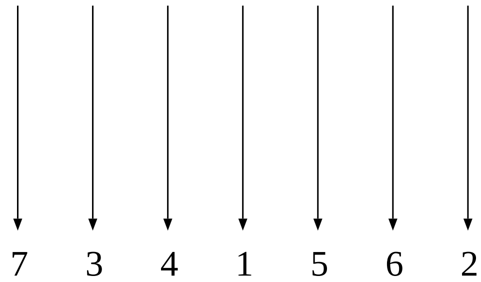
Amida: a kind of Budda

These *rays* mean "back light", which will be used as a lottery

Recent and Future Results (2010s～):

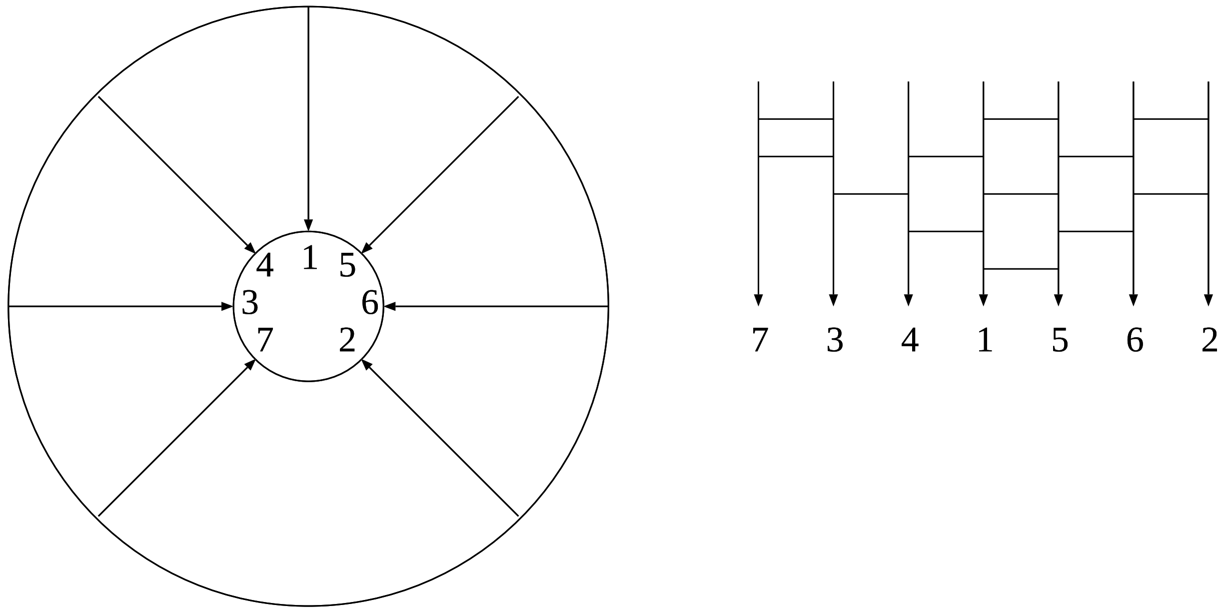
- More game-like, but not settled problem:
  - Amida-kuji (Ladder Lottery)





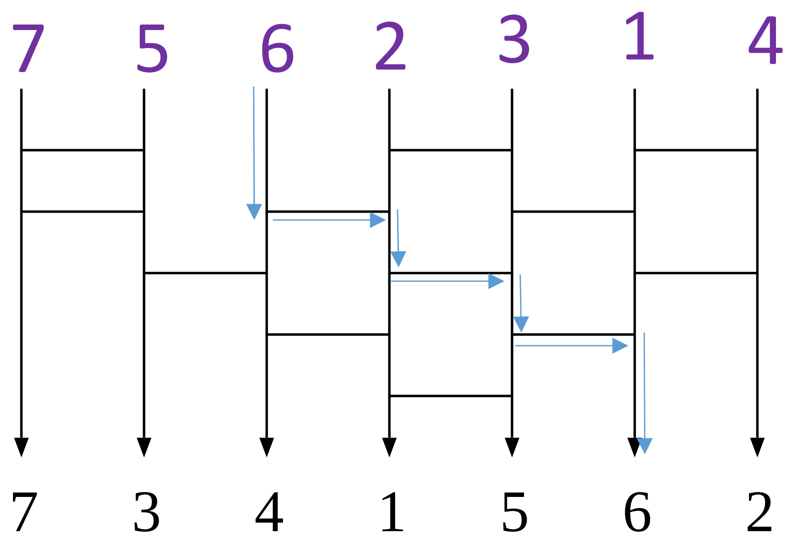
Recent and Future Results (2010s～):

- More game-like, but not settled problem:
  - Amida-kuji (Ladder Lottery)



Recent and Future Results (2010s～):

- More game-like, but not settled problem:
  - Amida-kuji (Ladder Lottery)



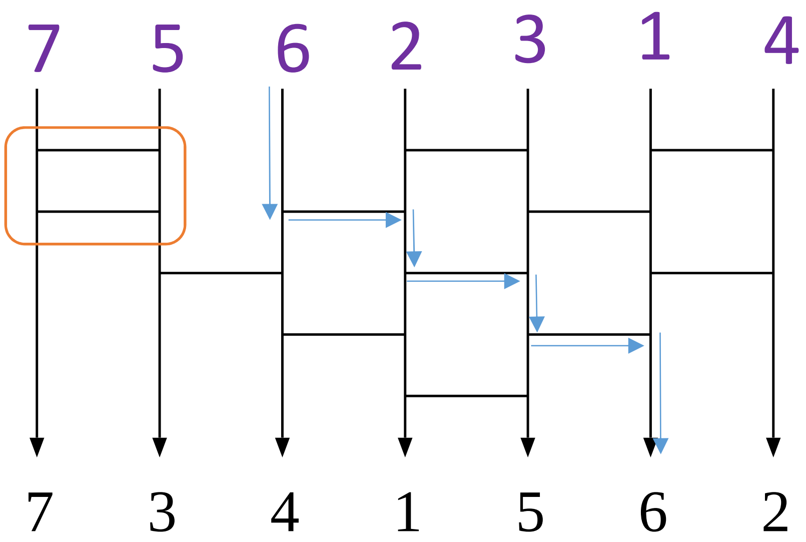
- Every Amida gives us a permutation.

Q: Why?

A: Every "bar" only swaps.

Recent and Future Results (2010s～):

- More game-like, but not settled problem:
  - Amida-kuji (Ladder Lottery)



- Every permutation, we can make an Amida.

Note: We can construct an *optimal* Amida with *fewest bars*.

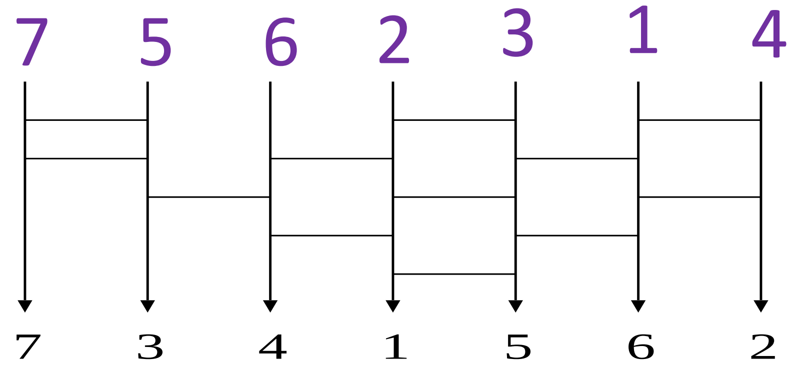
Q: How many?

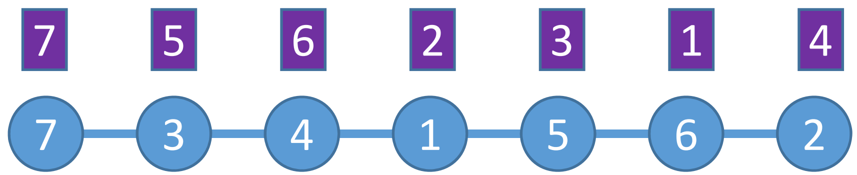
A: See the following paper:

K. Yamanaka, S.–i. Nakano, Y. Matsui, R. Uehara, and K. Nakada: Efficient Enumeration of All Ladder Lotteries and Its Application, *Theoretical Computer Science*, 2010.

Recent and Future Results (2010s～):

- More game-like, but not settled problem:
  - Collapsed Amida-kuji





Given: Path P with numbered vertices Numbered tokens on each vertex

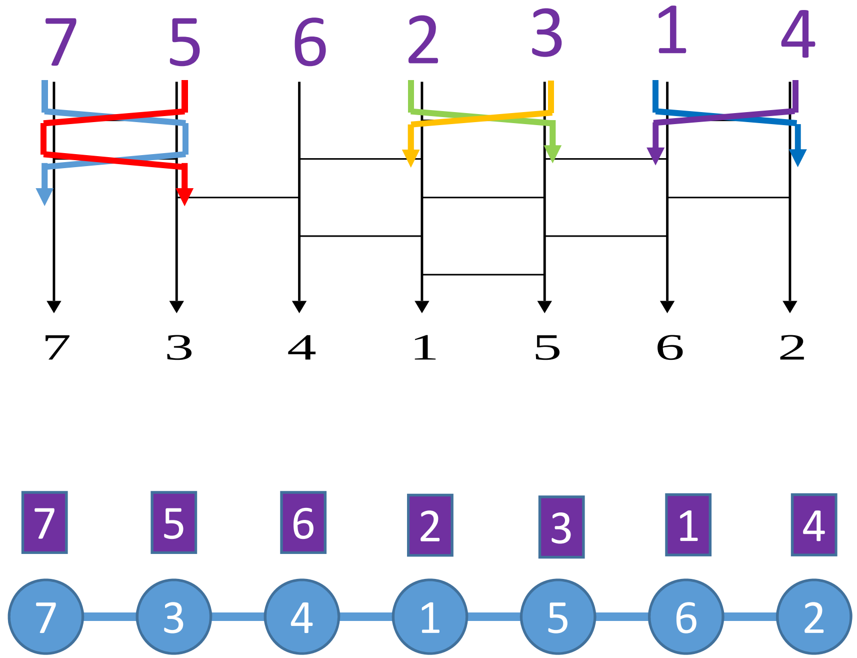
Rule: For an edge, swap both tokens

Goal: Arrange them ☺

Recent and Future Results (2010s～):

- More game-like, but not settled problem:
  - Collapsed Amida-kuji

Not necessarily *path*!



Given: Path P with
numbered vertices
Numbered tokens
on each vertex

Rule: For an edge, swap
both tokens

Goal: Arrange them ☺

Recent and Future Results (2010s～):

- More game-like, but not settled problem:
  - Generalized Amida-kuji

Yamanaka's idea!!

Given: Graph G with vertices 1, 2, …, n and tokens 1, 2, …, n on each vertex.

Rule: For an edge, swap both tokens

Goal: Put the token i on the vertex i

First presented at

K. Yamanaka, E. D. Demaine, T. Ito, J. Kawahara, M. Kiyomi, Y. Okamoto, T. Saitoh, A. Suzuki, K. Uchizawa, and T. Uno, Swapping Labeled Tokens on Graphs, in FUN with Algorithms, 2014.

Recent and Future Results (2010s～):

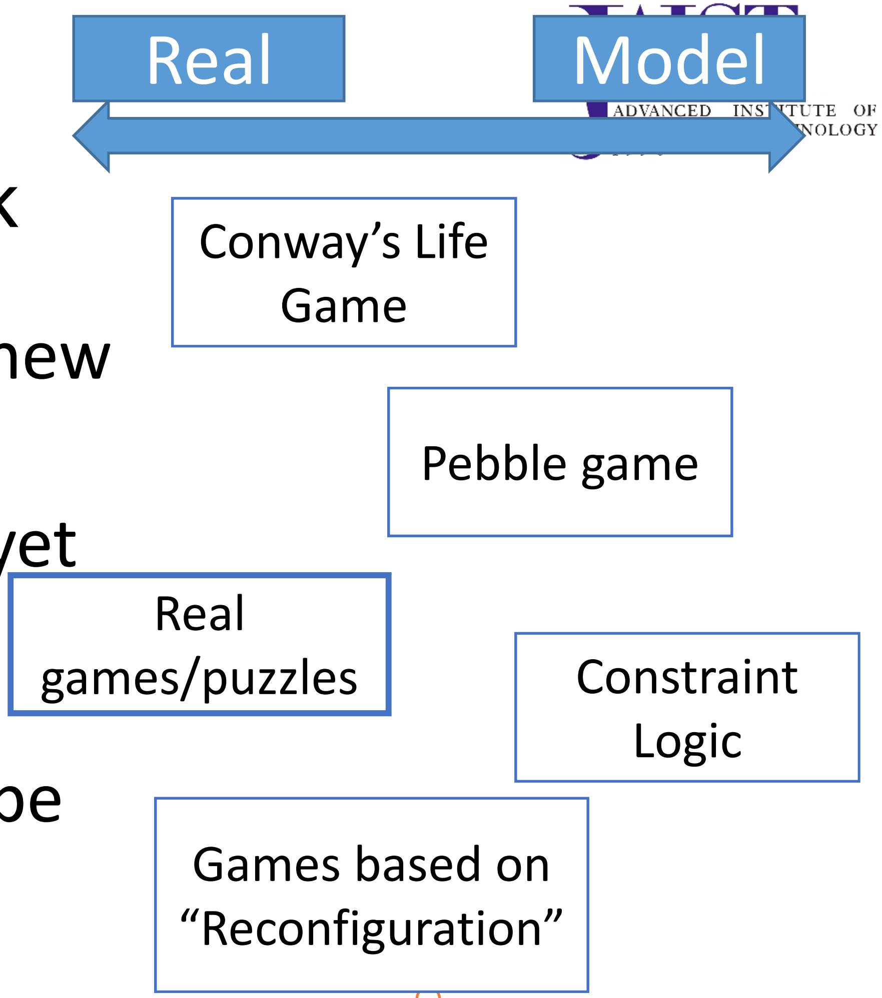
- More game-like, but not settled problem:
  - For Generalized Amida-kuji,
    - For any connected graph, we can solve it in $O(n^2)$ swapping, i.e., every instance is an yes instance.
    - Shortest path is quite difficult to find…
      - Path: $O(n^2)$ time
      - Cycle: Poly time
      - They show 4-approx. algorithm for a tree, but…?

No hardness results so far…

This seems to be (n^2-1) puzzle type problem.

K. Yamanaka, E. D. Demaine, T. Ito, J. Kawahara, M. Kiyomi, Y. Okamoto, T. Saitoh, A. Suzuki, K. Uchizawa, and T. Uno, Swapping Labeled Tokens on Graphs, in **FUN with Algorithms**, 2014.

Summary and Future work



- Games and Puzzles give us a new insight about "computation"
- Some new problems are not yet well-settled.
  - Reconfiguration problem, especially, (n^2-1) puzzle type problem.
  - We need new model that characterizes the classes P, NP, PSPACE, (EXP) in this manner.

These games are very much in want of a theory!