Regular Paper

NP-Completeness of Pandemic

Kenichiro Nakai[1]   Yasuhiko Takenaga[1,a]

Abstract: Pandemic is a multi-player board game which simulates the outbreak of epidemics and the human effort to prevent them. It is a characteristic of this game that all the players cooperate for a goal and they are not competitive. We show that the problem to decide if the player can win the generalized Pandemic from the given situation of the game is NP-complete.

Keywords: NP-completeness, game

1. Introduction

Much research have been conducted on the computational complexity of games. Games are generalized so that the games can be played on arbitrary large game boards and the problems to decide if the first player can win the game starting from the given situation of the game are considered. For example, it has been shown that Chess[1], Checkers[2], Go[3] and Shogi (or Japanese Chess)[4] are EXPTIME-complete and Reversi[5] and HEX[6] are PSPACE-complete. In addition to such traditional games, many new games have been created and some of them have become very popular. Among such games, it is shown that Scotland Yard is PSPACE-complete[7]. In this paper, we deal with a board game called Pandemic[8] and prove that generalized Pandemic is NP-complete.

Pandemic is a cooperative board game designed by Matt Leacoc and published in 2008. This game has won several famous awards given to excellent board games and is played very widely. Pandemic simulates the outbreak of epidemics and the human effort to prevent them. It is a characteristic of this game that all the players cooperate for a goal and they are not competitive. In this game, four epidemics have broken out in the world, each threatening to wipe out a region. The goal of the players is to discover cures for all the epidemics before it is too late.

2. Rules of Generalized Pandemic

In this section, we explain the rules of generalized Pandemic. We generalize Pandemic so that the underlying graph of the game board, the number of players and the number of epidemics can be arbitrary. Other rules are almost the same as usual Pandemic.

A situation of generalized Pandemic is given by the game board, cards in the hands of players and two piles of cards. The game board with s players and k epidemics is $B = (G, R, L, g, u, C, o)$, where $G = (V, E)$ is a graph which we call the *map* of the game, R is a collection of functions $r_a : V \rightarrow$ $\{0, 1, 2, 3\}(1 \leq a \leq k)$ that represents the infection rate of epidemic a at each vertex, $L \subseteq V$ is the set of vertices with a laboratory, $u : \{1, 2, \ldots, s\} \rightarrow V$ is a function that represents the vertex in which the pawn of each player is placed, $g : V \rightarrow \{1, 2, \ldots, k\}$ is a function that represents what is the epidemic each vertex is mainly infected by, $C \subseteq \{1, 2, \ldots, k\}$ is the set of epidemics whose cures are already discovered, $o \in \{0, 1, 2, 3, 4, 5, 6, 7\}$ is the number of occurred outbreaks. Let the set of vertices with the same value of g be the area of the epidemic. In the following, we call $u(i)$ the position of player i.

In addition, two kinds of cards are used in this game: infection cards and player cards. Both of them are piled face down when the game starts. There exist $|V|$ infection cards. Each infection card indicates a vertex of G, one card for each vertex. When the card indicating vertex v is drawn from the pile of infection cards, the infection rate $r_{g(v)}$ of vertex v is increased by one.

Player cards consist of city cards and epidemic cards. There exist $|V|$ city cards and four epidemic cards. Each city card indicates a vertex of G, one card for each vertex. City cards are used by the players in making actions as described below. Each player can keep seven city cards in his hand. If the number of cards in hand exceeds seven, the player must immediately discard arbitrary extra cards. Epidemic cards are special cards that increase the infection rate.

The game consists of turns of players. Player i plays in the $(sj + i)$-th turn ($j \geq 0$) for each $s = 0, 1, 2, \ldots$. The player who plays the turn is called the turn-player. A turn consists of three phases: an action phase, a draw phase and an infection phase, executed in this order. Actions that the turn-player can execute in an action phase are as follows.

- Drive: Move the player's pawn to an adjacent vertex.
- Direct Flight: Discard a city card from the player's hand and move the player's pawn to the vertex that the card indicates.
- Charter Flight: Discard the city card that indicates the position of the player and move to an arbitrary vertex when he has the card.
- Shuttle Flight: Move the player's pawn to any other vertex with a laboratory when the player's pawn is on a vertex with

[1] The University of Electro-Communications, Chofu, Tokyo 182–8585, Japan
[a] takenaga@cs.uec.ac.jp

a laboratory.

- Build a Laboratory: Discard the city card indicating the player's position and build a laboratory on the vertex when he has the card.
- Share Knowledge: Transfer the city card indicating the player's position to another player when the turn-player and the other player are in the same vertex and the turn-player has the card.
- Discover a Cure: Discard five city cards indicating the vertices with the same value of g to discover the cure of the corresponding epidemic when the player is on the vertex with a laboratory.
- Treat Disease: Decrease by one the infection rate of the vertex of the player's position.

In an action phase, the turn-player executes four actions. Note that the same actions can be made more than once in one turn.

In a draw phase, the turn-player draws two player cards from the pile and adds city cards to his hand. If the player draws an epidemic card, take the bottom card from the pile of infection cards and increase by three the infection rate $r_{g(v)}$ of city v indicated by the card. Then shuffle all the normal infection cards which are already drawn and put them onto the pile of infection cards.

In an infection phase, two infection cards are drawn from the pile and increase by one the infection rate $r_{g(v)}$ of each vertex v indicated by the cards. When the infection rate of epidemic a is increased at vertex v with infection rate $r_a(v) = 3$, an *outbreak* occurs. If an outbreak occurs, instead of increasing the infection rate of v, for each vertex u adjacent to v, increase the infection rate $r_a(u)$ by one. Though infection cards indicating city v only increases the infection rate of the epidemic by which v is mainly infected, when an outbreak occurs, cities that are not in the area of the epidemic $g(v)$ can be infected by epidemic $g(v)$. An outbreak in vertex v may cause another outbreak in its adjacent vertex, causing a *chain reaction*. Note that each vertex causes an outbreak at most once in each chain reaction.

All the players lose the game when outbreaks occur eight times or when there remains no card to draw in a draw phase. All the players win if cures for all k epidemics are discovered before then.

3. NP-completeness of Generalized Pandemic

We define Generalized Pandemic Problem (GPP for short) as a problem to decide if players can win from a given situation of the Generalized Pandemic when all the cards to be drawn from the piles are given as a part of the input, including the infection cards put onto the pile after an epidemic card is drawn.

Theorem 3.1 GPP is NP-complete for any number of players.

It is easy to show that $GPP \in NP$. As the game ends when there remains no player card to draw in a draw phase and two player cards are drawn in one turn, the game ends in at most $(|V| + 4)/2 + 1 = |V|/2 + 3$ turns. Thus, given the play history of Pandemic, we can check if the players win or lose in polynomial time. Therefore, $GPP \in NP$.

We prove the NP-hardness of GPP in the following of this section.

3.1 Reduction

At first, we prove NP-hardness of the case when $p = 1$, that is the case of a single player. We show later that GPP is NP-hard for any number of players. We give a reduction from the Hamiltonian cycle problem [9].

Let $H = (V', E')$ be an instance of the Hamiltonian cycle problem. Let $|V'| = n$. In the following, we call V' the set of nodes in order to distinguish them from the vertices of the map of GPP.

The outline of the reduction is as follows. We transform a node of H to a gadget including a vertex which is about to make an outbreak. The player must visit all such vertices, go back to the node where he started and discover a cure in a fixed number of turns. Otherwise the unvisited vertex will outbreak, causing a game over.

3.1.1 Game Board

The instance of Generalized Pandemic obtained from H is as follows. First, we explain how to construct the game board of GPP. We now assume that n is even. We call vertex v which satisfies $r_{g(v)}(v) = 3$ an *alert point*. Any vertex which is not an alert point has infection rate 0 for any epidemic and is called a *normal vertex*. From the order of infection cards in the pile, which we explain later, outbreaks can occur at any alert point unless the player makes actions to prevent them. We call the action to decrease the infection rate of an alert point as the *release* of the alert point.

An arbitrary node of H is replaced by *the starting node gadget* and all the other nodes of H are replaced by *node gadgets*. *Edge gadgets* are placed on all the edges between the starting node gadget and adjacent node gadgets.

An example of a graph H and the map of GPP that is obtained from H by our reduction are shown in **Fig. 1** and **Fig. 2**.
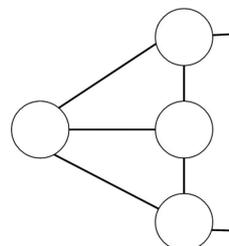
Fig. 1   An instance H of the Hamiltonian cycle problem.
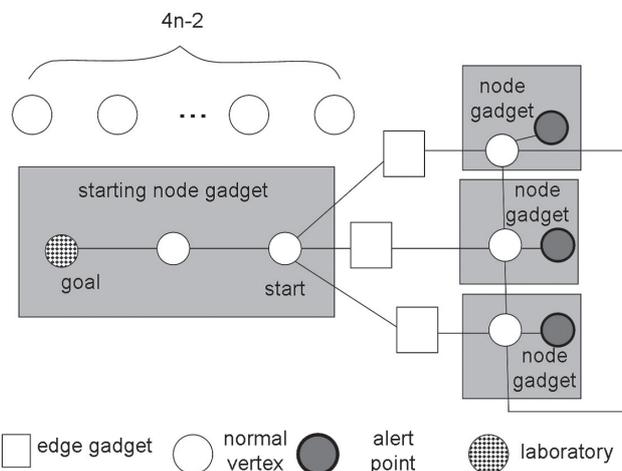
Fig. 2   Game board of BPP obtained from H.
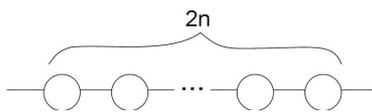
Fig. 3 An edge gadget.

Now we explain the detail of the above gadgets. A node gadget consists of two vertices as shown in Fig. 2. One vertex is an alert point and the other is a normal vertex. A starting node gadget is as shown in Fig. 2. The vertices called the start vertex and the goal vertex are in the starting node gadget. The player's pawn is on the start vertex and there exists a laboratory in the goal vertex. To win the game, the player must release all the alert points in the node gadgets and then go to the goal vertex to discover a cure.

An edge gadget is a path with $2n$ normal vertices as shown in **Fig. 3**. Edge gadgets are placed on the edges between the starting node gadget and adjacent node gadgets. The leftmost vertex of the gadget is connected to the start vertex and the rightmost vertex is connected to the normal vertex of a node gadget. If H has a Hamiltonian cycle and the player makes appropriate moves, infection cards of the alert points in the node gadgets are drawn from the pile when the player is on an edge gadget to go back to the starting node gadget. Thus, if an alert point in a node gadget is not released, an outbreak occurs before the cure is discovered.

There exist two epidemics 1 and 2 in the game board and $1 \notin C$ and $2 \in C$ hold. That is, there remains only one epidemic whose cure is not discovered. $g(v) = 1$ holds for five isolated vertices and $g(v) = 2$ for all the other vertices. The number of occurred outbreaks is seven.

3.1.2 Cards

The piles of cards obtained from our reduction is as follows.

An infection card that indicates an alert point is called a dangerous infection card. An infection card which is not dangerous is called a safe infection card. The pile of infection cards consists of $3n - 2$ safe infection cards, $n - 1$ dangerous infection cards and one safe infection card from top to bottom. Dangerous infection cards are drawn from the pile between the $3n/2$-th turn and $(2n - 1)$-st turn. No dangerous infection cards are drawn before the turns.

The player has no card in his hand. The pile of player cards consists of $4n - 7$ city cards indicating isolated vertices that satisfy $g(v) = 2$ and five city cards indicating isolated vertices that satisfy $g(v) = 1$ from top to bottom.

It is obvious that the game board and the piles of cards can be constructed in logarithmic space.

3.2 Correctness of the Reduction

In this section, we prove the correctness of the reduction.

First, we prove that the player can win the game if H has a Hamiltonian cycle. If H has a Hamiltonian cycle, the player, starting from the start vertex, visits the vertex gadgets in the order corresponding to the Hamiltonian cycle and releases the alert points. Then the player goes to the goal vertex and discovers the cure.

We show that the above actions can be completed before the player loses. It takes $n/2$ turns to go to the right end of an edge gadget. For each node gadget, the player needs one turn; go to an adjacent node gadget, go to the alert point, release the alert point and go back to the normal vertex in the gadget. Thus, it takes $n/2 + (n - 1) = 3n/2 - 1$ turns to release all the alert points in the node gadgets. As the dangerous infection cards are drawn after the $(3n/2 - 1)$-st turn, no outbreak occurs at the alert points. After that, $n/2$ rounds are needed to go through another edge gadget, and one more round is needed to go back to the goal vertex and discover the cure. Since the player has five player cards corresponding to the vertices with epidemic 1 after the draw phase of the $(2n - 1)$-st turn, the player can discover the cure in the $2n$-th turn. Though the pile of the player cards becomes empty after the draw phase of the $(2n - 1)$-st turn, the player can discover the cure before the next draw phase. Therefore, if H has a Hamiltonian cycle, the player can win the game.

Next, we prove that the player cannot win if H does not have a Hamiltonian cycle. To begin with, observe that the player can only use drive actions to move on the map to win the game. As all the player cards that the player draws indicate isolated vertices, obviously a charter flight action cannot be executed, and a shuttle flight action cannot be executed because no new laboratory can be built. Also, even if the player uses a direct flight action, the player can go only to isolated vertices and cannot go back to the other gadgets.

In order to discover the cure for epidemic 1, the player must go to the goal before the $2n$-th turn. Thus the player must be in the normal vertex of a node gadget which is connected to an edge gadget at the end of the $(3n/2 - 1)$-st turn. Clearly, if H does not have a Hamiltonian cycle, the player cannot go to the vertex after releasing all the alert points in the node gadgets. If an alert point in a node gadget is not released before the $(2n - 1)$-st turn, as the player has to go to the goal, an outbreak occurs at the alert point and the player loses. Therefore, if H does not have a Hamiltonian cycle, the player cannot win the game.

With a slight modification, the above proof can be applied to the case when n is odd. It is sufficient to adjust the length of edge gadgets and the number of cards in the piles so that it takes $4\lceil n/2 \rceil$ drive actions to go through an edge gadget.

From above, the correctness of our reduction is proved. Therefore, it is shown that GPP is NP-complete.

3.3 Discussions

First, we extend the proof for the case of multiple players. To the above construction, we add players in a way that they cannot be concerned with discovering cures. By locating the added players on added isolated vertices, they can make no useful actions.

Next, we take the rules into consideration that we have not considered in our definition of GPP for simplicity. In real Pandemic, each player has one of five occupations. Each player has a special ability depending upon their occupation. We can see that even if the players have special abilities, and there must exist at least one player for each occupation, GPP remains NP-complete. Among the special abilities, that of the dispatcher requires a little modification of our reduction. A dispatcher can move another player's pawn on his turn as if it were his own pawn. He can also move any pawn to any vertex where another pawn is placed. Our reduc-

tion must be modified so that each player except player 1, who is in the start vertex, is located in an end vertex of a path of length $8n - 4$, not an isolated vertex. The other end of the path is an alert point and an outbreak can occur in the $(2n - 1)$-st turn. Then, to release the alert point, each player except player 1 must drive to the alert point and has no time to use their special ability.

4. Conclusion

In this paper, we have proved that Generalized Pandemic is NP-complete. Additionally, with most of the expansion rules, Generalized Pandemic is still NP-complete. In an additional package for Pandemic, the rule called bio-terrorist challenge is introduced. Under this rule, a player called the bio-terrorist plays against the other players. Thus the problem to decide if the bio-terrorist or the other players can win from the given situation is probably PSPACE-complete because the bio-terrorist and the other players play to beat each other.

References

[1] Fraenkel, A.S. and Lichtenstein, D.: Computing a Perfect Strategy for $n \times n$ Chess Requires Time Exponential in n, *J. Combinatorial Theory*, Ser. A, Vol.31, No.2, pp.199–214 (1981).
[2] Robson, T.M.: *N* by *N* Checkers is EXPTIME Complete, *SIAM J. Comput.*, Vol.13, No.2, pp.252–267 (1984).
[3] Robson, T.M.: The Complexity of Go, *Proc. IFIP*, pp.413–417 (1983).
[4] Adachi, A., Kamekawa, H. and Iwata, S.: Shogi on $n \times n$ board is Complete in Exponential Time, *Trans. IEICE*, Vol.J70-D, pp.1843–1852 (1987) (in Japanese).
[5] Iwata, S. and Kasai, T.: The Othello Game on an $n \times n$ Board is PSPACE-Complete, *Theoret. Comput. Sci.*, Vol.123, No.2, pp.329–340 (1994).
[6] Reisch, S.: Hex ist PSPACE-vollständig, *Acta Inf.*, Vol.15, pp.167–191 (1981).
[7] Sevenster, M.: The Complexity of Scotland Yard, Research Report, Institute for Logic, Language and Computation, University of Amsterdam, PP-2006-18 (2006).
[8] Z-Man Games, (online), available from ⟨http://www.zmangames.com/⟩ (accessed 2011-08-30).
[9] Garey, M.R. and Johnson, D.S.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman & Co., New York (1979).

Yasuhiko Takenaga was born in 1966. He received his B.E., M.E. and Ph.D. degrees in information science from Kyoto University in 1989, 1991 and 1995, respectively. He became an instructor at Kyoto University in 1991. He joined the University of Electro-Communications in 1997 as an assistant professor and is currently an associate professor. His research interests include graph algorithms, graph representation of Boolean functions, and the complexity of games and puzzles.

Kenichiro Nakai was born in 1986. He received a B.E. degree from the Department of Information Science, at the University of Electro-Communications, Tokyo, in 2010. He is currently with the Graduate School of Informatics and Engineering, at the University of Electro-Communications. His research interests include algorithms and the complexity of games and puzzles.