LETTER

# Generalized Pyramid is NP-Complete*

**Chuzo IWAMOTO**[†a)], ***Member and*** **Yuta MATSUI**[†], ***Nonmember***

**SUMMARY** Pyramid is a solitaire game, where the object is to remove all cards from both a pyramidal layout and a stock of cards. Two exposed cards can be matched and removed if their values total 13. Any exposed card of value 13 and the top card of the stock can be discarded immediately. We prove that the generalized version of Pyramid is NP-complete.

***key words:*** *NP-complete, computational complexity, one-player game, pyramid*

## 1. Introduction

Pyramid is a solitaire game, where the object is to remove all cards from both a pyramidal layout and a stock of cards (see Fig. 1). The pyramidal layout is composed of seven layers, where seven cards of the first layer cover six cards of the second layer, which further cover five cards of the third layer, and so on. The remaining cards are piled up, which are the stock.

A card is *exposed* if no cards cover it. Two exposed cards can be matched and removed if their values total 13. Initially, seven cards in the first layer and the top card of the stock are exposed. The cards of value 13 (Kings) can be removed immediately if they are exposed. The top card of the stock can be matched only with an exposed card in the pyramid. If no match is made, then the top card of the stock can be discarded. Once the stock is exhausted or no more pairs can be made, the game ends. The aim of the game is to remove all of the cards of the pyramid and stock.

In Fig. 1, two cards ♢6 and ♣7, a card ♡13, and two cards ♢4 and ♠9 can be removed. After the removal of five cards, ♣11 and ♠2 can be removed, but ♢3 and ♠10 cannot be removed, since ♠10 is not exposed. At this point, there are no pair of cards totaling 13. Fortunately, if the top card ♠12 of the stock is discarded, then ♣5 matches ♠8.

In this paper, we consider the generalized version of Pyramid. The *generalized* 4*k-card deck* includes $k$ ranks of each of the four suits, spades (♠), hearts (♡), diamonds (♢), and clubs (♣). A card $r \in \{1, 2, \ldots, \lfloor k/2 \rfloor\}$ matches a card $k - r$ when $k$ is odd, while a card $r \in \{1, 2, \ldots, k/2\}$ matches a card $k - r + 1$ when $k$ is even. The instance of the *Generalized Pyramid Problem* is the initial layout of cards consisting of an $l$-layer pyramid and a stock of $s$ cards, where $s = 4k - l(l+1)/2$. The problem is to decide whether the player can remove all of the $4k$ cards from the initial layout.

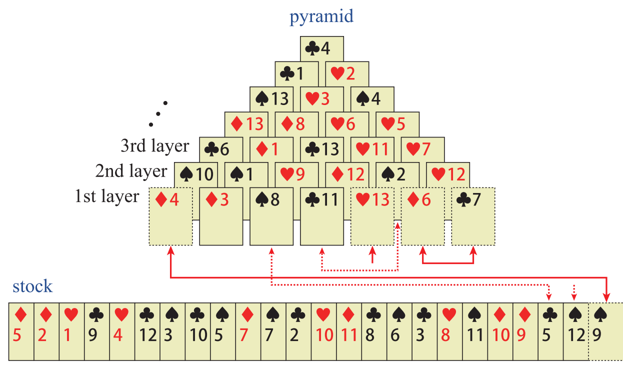


**Fig. 1** Initial layout of pyramid.

We will show that the Generalized Pyramid Problem is NP-complete, even if the number $s$ of stock cards satisfies $s \in \{1, 2, 3, 4\}$. (Note that if $s = 0$, the player never removes the last two cards.) It is not difficult to show that the Generalized Pyramid Problem is in NP, since the player can remove at most $4k$ cards.

There has been a huge amount of literature on the computational complexities of games and puzzles. In 2009, a survey of games, puzzles, and their complexities was reported by Hearn and Demaine [8]. Recently, Block Sum [7], Hashiwokakero [1], Kaboozle [2], Kurodoko [10], Magnet Puzzle [11], Pandemic [12], and Zen Puzzle Garden [9] were shown to be NP-complete. Furthermore, it is known that a single-player (resp. two-player) version of UNO is NP-complete (resp. PSPACE-complete) [5], and Rolling Block Maze is PSPACE-complete [3].

## 2. Reduction from 3SAT to Generalized Pyramid

### 2.1 3SAT Problem

The definition of 3SAT is mostly from [6]. Let $U = \{x_1, x_2, \ldots, x_n\}$ be a set of Boolean *variables*. Boolean variables take on values 0 (false) and 1 (true). If $x$ is a variable in $U$, then $x$ and $\overline{x}$ are *literals* over $U$. The value of $\overline{x}$ is 1 (true) if and only if $x$ is 0 (false). A *clause* over $U$ is a set of literals over $U$, such as $\{\overline{x_1}, x_3, x_4\}$. It represents the disjunction of those literals and is *satisfied* by a truth assignment if and only if at least one of its members is true under that assignment.

Manuscript received May 10, 2013.
Manuscript revised July 2, 2013.
[†]The authors are with the Graduate School of Engineering, Hiroshima University, Higashihiroshima-shi, 739–8527 Japan.
*This research was supported in part by Scientific Research Grant, Ministry of Japan.
a) E-mail: chuzo@hiroshima-u.ac.jp
DOI: 10.1587/transinf.E96.D.2462

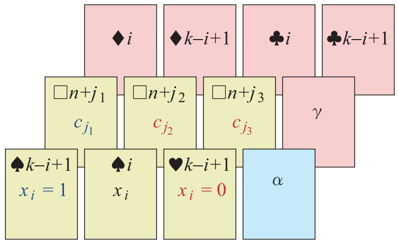


**Fig. 2** $3 \times 4$ cards transformed from variable $x_i$.

An instance of 3SAT is a collection $C = \{c_1, c_2, \ldots, c_m\}$ of clauses over $U$ such that $|c_j| \leq 3$ for each $c_j \in C$. The 3SAT problem asks whether there exists some truth assignment for $U$ that simultaneously satisfies all the clauses in $C$. This problem is known to be NP-complete. For example, $U = \{x_1, x_2, x_3, x_4\}$, $C = \{c_1, c_2, c_3, c_4\}$, and $c_1 = \{x_1, x_2, \overline{x_3}\}$, $c_2 = \{\overline{x_1}, \overline{x_2}, x_4\}$, $c_3 = \{\overline{x_1}, x_3, \overline{x_4}\}$, $c_4 = \{\overline{x_2}, \overline{x_3}, \overline{x_4}\}$ provide an instance of 3SAT. For this instance, the answer is "yes", since there is a truth assignment $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$ satisfying all clauses. It is known that 3SAT is NP-complete even if each variable occurs exactly once positively and exactly twice negatively in $C$ [4].

### 2.2 Transformation from an Instance of 3SAT to an Initial Layout of Cards

We present a polynomial-time transformation from an arbitrary instance $C$ of 3SAT to an initial layout of cards such that $C$ is satisfiable if and only if all cards can be removed.

Let $n$ and $m$ be the numbers of variables and clauses of $C$, respectively. The generalized $4k$-card deck includes $k$ ranks of each of the four suits. In the proof, we assume that $k$ is an even number. In this case, a card of rank $r$ matches a card of rank $k-r+1$ for each $r \in \{1, 2, \ldots, k/2\}$. (If $k$ is odd, then a card $r$ matches $k - r$ for each $r \in \{1, 2, \ldots, \lfloor k/2 \rfloor\}$. In this case, the following transformation is modified so that the four cards of rank $k$ are used as A, A′, B, B′ in Fig. 4.)

Each variable $x_i \in \{x_1, x_2, \ldots, x_n\}$ is transformed into $3 \times 4$ cards shown in Fig. 2. (See also Fig. 4 when $n = m = 4$ and $k = 102$. The $3 \times 4$ cards for $x_1$ are $\spadesuit 102$, $\spadesuit 1$, $\heartsuit 102$, $\alpha$; $\spadesuit 5$, $\spadesuit 6$, $\spadesuit 7$, $\gamma$; and $\diamondsuit 1$, $\diamondsuit 102$, $\clubsuit 1$, $\clubsuit 102$.)

In Fig. 2, the value of $k$ depends on $n$ and $m$; we will fix $k$ later. The first three cards of the first layer of Fig. 2 are $\spadesuit k-i+1$, $\spadesuit i$, and $\heartsuit k-i+1$ (and they are labeled with "$x_i = 1$", "$x_i$", and "$x_i = 0$", respectively). If variable $x_i$ appears in clause $c_{j_1}$ positively and in $c_{j_2}$ and $c_{j_3}$ negatively, then the first three cards of the second layer have ranks $n+j_1$, $n+j_2$, and $n+j_3$, respectively. If this is the first (resp. second, third) appearance of a card of rank $n+j$, the suit is $\spadesuit$ (resp. $\heartsuit$, $\diamondsuit$). (For example, $\spadesuit 5$, $\heartsuit 5$, and $\diamondsuit 5$ labeled with $c_1$ appears in that order in the second layer of Fig. 4, since variable $x_1$ appears clauses $c_1$, $c_2$, and $c_3$ in that order.)

Blue and red cards of Fig. 2 are dummy so that the set of cards forms a $3 \times 4$ layout. (Blue (resp. red) dummy cards will be arranged so that they can be removed trivially at the start (resp. end) of the game. We will explain them later.)

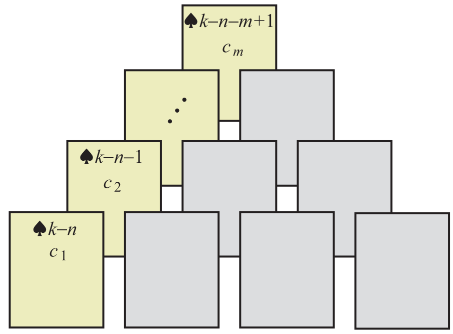


**Fig. 3** An $m$-layer pyramid transformed from clauses $c_1, c_2, \ldots, c_m$.

If $\spadesuit i$ is matched with $\spadesuit k-i+1$, then card $n+j_1$ (labeled with $c_{j_1}$) will be exposed. If $\spadesuit i$ is matched with $\heartsuit k-i+1$ (and if the blue dummy card has already been removed), then cards $n+j_2$ and $n+j_3$ (labeled with $c_{j_2}$ and $c_{j_3}$) are exposed. Later, one can see that matching $\spadesuit i$ with $\spadesuit k-i+1$ (resp. $\heartsuit k-i+1$) implies the assignment $x_i = 1$ (resp. $x_i = 0$).

The set of clauses $\{c_1, c_2, \ldots, c_m\}$ is transformed into an $m$-layer pyramid shown in Fig. 3, where cards labeled with $c_m, \ldots, c_2, c_1$ are piled up. Grey cards are dummy so that $m(m+1)/2$ cards form a pyramidal layout. It should be noted that card $\spadesuit k-n-m+1$ (labeled with $c_m$) can be removed only if all of the $m-1$ cards $\spadesuit k-n, \spadesuit k-n-1, \ldots, \spadesuit k-n-m+2$ (labeled with $c_1, c_2, \ldots, c_{m-1}$) have been removed.

Figure 4 is the bottom $n+m+1$ layers of the initial layout of the pyramid transformed from $C = \{c_1, c_2, c_3, c_4\}$, where $c_1 = \{x_1, x_2, \overline{x_3}\}$, $c_2 = \{\overline{x_1}, \overline{x_2}, x_4\}$, $c_3 = \{\overline{x_1}, x_3, \overline{x_4}\}$, and $c_4 = \{\overline{x_2}, \overline{x_3}, \overline{x_4}\}$. The $m$-layer pyramid of Fig. 3 is followed by a pile of cards

$$\heartsuit 1, \heartsuit 2, \ldots, \heartsuit n$$

so that there are four cards $\{\spadesuit r, \heartsuit r, \diamondsuit r, \clubsuit r\}$ for each rank $r \in \{1, 2, \ldots, n\}$ (see Fig. 4). For the same reason, (i) the cards of rank $r \in \{n+1, n+2, \ldots, n+m\}$ not used in the second layer and (ii) the following $3m$ cards are piled up.

$$\begin{aligned}&\heartsuit k-n,\ \heartsuit k-n-1,\ \ldots,\ \heartsuit k-n-m+1,\\ &\diamondsuit k-n,\ \diamondsuit k-n-1,\ \ldots,\ \diamondsuit k-n-m+1,\\ &\clubsuit k-n,\ \clubsuit k-n-1,\ \ldots,\ \clubsuit k-n-m+1.\end{aligned}$$

In Fig. 4, the cards of (i) are $\{\clubsuit 5, \clubsuit 6, \clubsuit 7, \clubsuit 8\}$. The cards of (i) and (ii) are piled up so that they can be removed trivially at the end of the game. (In Fig. 4, those cards are piled in the following order: $\heartsuit 98$, $\heartsuit 97$, $\heartsuit 96$, $\heartsuit 95$; $\diamondsuit 98$, $\diamondsuit 97$, $\diamondsuit 96$, $\diamondsuit 95$; and $\clubsuit 98$, $\clubsuit 97$, $\clubsuit 96$, $\clubsuit 95$. We will explain them later.)

In Fig. 4, blue cards are labeled with $\alpha, \alpha', \beta, \beta', \cdots$. For every $\chi \in \{\alpha, \beta, \ldots\}$, blue cards $\chi$ and $\chi'$ have such ranks that $\chi$ matches $\chi'$. Thus, every such pair can be removed at the start of the game. (Without loss of generality, we can assume that the number $n$ of variables is even.)

Grey cards are labeled with A, A′, B, B′, …, where card A matches A′, card B matches B′, and so on. All gray cards except {S,T} can be removed trivially at the start of the game. (Figure 5 is the layout after the removal of those blue and gray cards.)

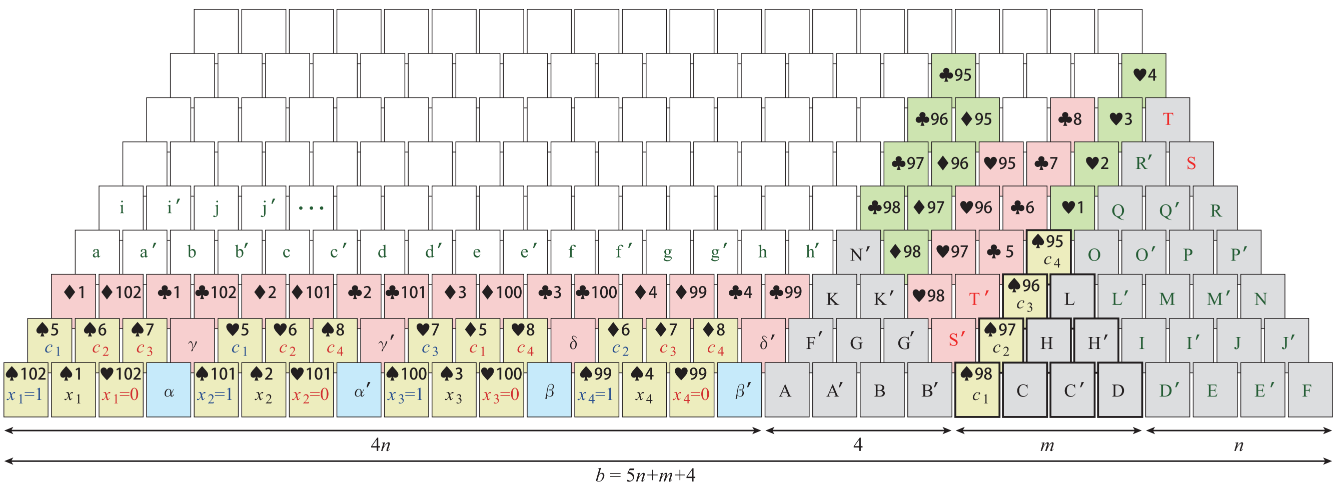


**Fig. 4** The bottom $n+m+1$ layers of the initial layout of the pyramid transformed from $C = \{c_1, c_2, c_3, c_4\}$, where $c_1 = \{x_1, x_2, \overline{x_3}\}$, $c_2 = \{\overline{x_1}, \overline{x_2}, x_4\}$, $c_3 = \{\overline{x_1}, x_3, \overline{x_4}\}$, and $c_4 = \{\overline{x_2}, \overline{x_3}, \overline{x_4}\}$. A pair of cards of rank $u$ and $v$ match if $u + v = k + 1$ (= 103).

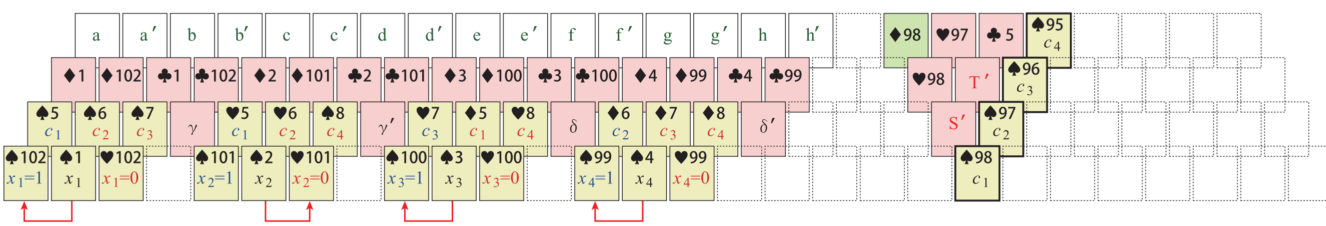

**Fig. 5** Assignment $(x_1, x_2, x_3, x_4) = (1, 0, 1, 1)$.

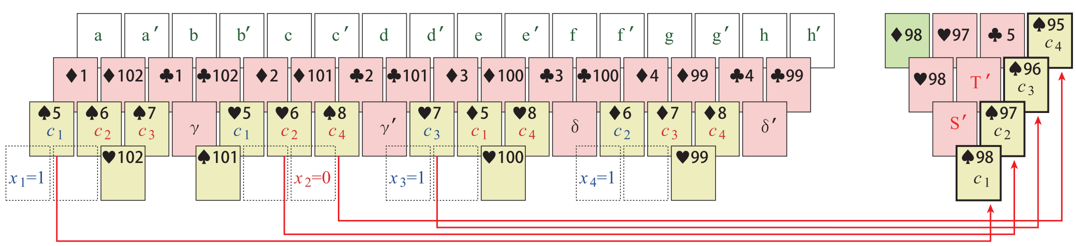

**Fig. 6** All clauses $c_1, c_2, \ldots, c_m$ are satisfied if and only if at least one of the three literals of every clause $c_j \in \{c_1, c_2, \ldots, c_m\}$ has value 1.

The number $b$ of cards in the bottom layer is $b = 5n + m + 4$ (= 28 when $n = m = 4$). Thus, the total number $p$ of cards of the pyramid is $p = b(b+1)/2$ (= 406). Let $k'$ be the integer such that $p = 4k' + s'$, where $s' \in \{0, 1, 2, 3\}$. Now we fix the value of $k$ as $k = (p + (4 - s'))/4$ (= 102). The number $s$ of cards in the stock is $s = 4k - p$ (= 2). We can match a pair of cards of rank $u$ and $v$ if $u + v = k + 1$ (= 103).

## 2.3 NP-Completeness of Generalized Pyramid

In this section, we will show that the instance $C$ of 3SAT is satisfiable if and only if all the cards of the pyramid and the stock can be removed.

Assume that the instance $C$ of 3SAT is satisfiable. Each card of label "$x_i$" can be matched with either card "$x_i = 1$" or "$x_i = 0$" (see Fig. 5). If we match card "$x_i$" with "$x_i = 1$", then card $c_{j_1}$ is exposed. If we match card "$x_i$" with "$x_i = 0$", then cards $c_{j_2}$, $c_{j_3}$ are exposed. (Recall that we assumed that $x_i$ appears in clause $c_{j_1}$ positively and in $c_{j_2}$, $c_{j_3}$ negatively.)

Since $C$ is satisfiable, we can remove card "$x_i$" so that at least one of the three $c_j$-cards of rank $n + j$ (in the second layer) is exposed for every $j \in \{1, 2, \ldots, m\}$ (see Fig. 5). Therefore, all of the cards $c_1, c_2, \ldots, c_m$ of Fig. 3 can be removed (see Fig. 6).

Once all those $m$ cards are removed, we can trivially remove pairs of dummy cards S, S′ and T, T′ of Fig. 4. Then, we remove all pairs of green cards of rank $r \in \{1, 2, \ldots, n\}$ and the remaining yellow cards in the first layer (for example, ♡1 and ♡102 in Figs. 4 and 6, respectively). Finally, we trivially remove (i) all pairs of red cards in the second layer, (ii) all pairs of red ♡-cards and red ♣-cards, (iii) all pairs of

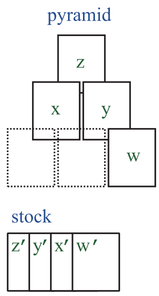


**Fig. 7** The last four cards and the stock when $s = 4$. The cases when $s \in \{1, 2, 3\}$ are similar and omitted.

the remaining green cards and yellow cards, (iv) all pairs of red cards in the third layer, and (v) all pairs of white cards. (For example, (i) $\gamma$ and $\gamma'$, (ii) ♡98 and ♣5, (iii) ♢98 and ♡5, (iv) ♢1 and ♢102, and (v) a and a′ in Figs. 4 and 6.) The last $s$ of the white cards are removed by using the stock (see Fig. 7). Hence, if the instance $C$ of 3SAT is satisfiable, then all the cards of the pyramid and the stock can be removed.

Assume that the player can remove all the cards of the pyramid and the stock. The card ♡1 (see Fig. 4) can be removed only if all the $m$ cards labeled with $c_1, c_2, \ldots, c_m$ of Fig. 3 can be removed. Those $m$ cards can match yellow cards in the second layer (see Fig. 6). Any yellow card in the second layer is exposed only if the corresponding pair of yellow cards in the first layer are removed. The set of pairs of cards removed from the first layer indicates the truth assignment satisfying all clauses of $C$. (From Fig. 6, one can see that $x_1 = 1$, $x_2 = 0$, and $x_3 = 1$ satisfy all the clauses of $C = \{c_1, c_2, c_3, c_4\}$.) This completes the proof.

## References

[1] D. Andersson, "Hashiwokakero is NP-complete," Inf. Process. Lett., vol.109, pp.1145–1146, 2009.

[2] T. Asano, E.D. Demaine, M.L. Demaine, and R. Uehara, "NP-completeness of generalized Kaboozle," J. Inf. Process., vol.20, no.3, pp.713–718, 2012.

[3] K. Buchin and M. Buchin, "Rolling block mazes are PSPACE-complete," J. Inf. Process., vol.20, no.3, pp.719–722, 2012.

[4] M.R. Cerioli, L. Faria, T.O. Ferreira, C.A.J. Martinhon, F. Protti, and B. Reed, "Partition into cliques for cubic graphs: planar case, complexity and approximation," Discrete Appl. Math., vol.156, no.12, pp.2270–2278, 2008.

[5] E.D. Demaine, M.L. Demaine, R. Uehara, T. Uno, and Y. Uno, "UNO is hard, even for a single player," Proc. Fun with Algorithms (Lect. Notes Comput. Sci.), vol.6099, pp.133–144, 2010.

[6] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman, New York, NY, USA, 1979.

[7] K. Haraguchi and H. Ono, "BlockSum is NP-complete," IEICE Trans. Inf. & Syst., vol.E96-D, no.3, pp.481–488, 2013.

[8] R.A. Hearn and E.D. Demaine, Games, Puzzles, and Computation, A.K. Peters Ltd., 2009.

[9] R. Houston, J. White, and M. Amos, "Zen puzzle garden is NP-complete," Inf. Process. Lett., vol.112, pp.106–108, 2012.

[10] J. Kölker, "Kurodoko is NP-complete," J. Inf. Process., vol.20, no.3, pp.694–706, 2012.

[11] J. Kölker, "The magnet puzzle is NP-complete," J. Inf. Process., vol.20, no.3, pp.707–708, 2012.

[12] K. Nakai and Y. Takenaga, "NP-completeness of Pandemic," J. Inf. Process., vol.20, no.3, pp.723–726, 2012.