数理解析研究所講究録
第2067巻 2018年 26-36

数式処理システムMathematicaを用いた高校生による数学研究 III

兵庫教育大学 福井 昌則 (Masanori Fukui)
Hyogo University of Teacher Education
関西学院高等部 宮寺 良平 (Ryohei Miyadera)
Kwansei Gakuin High School

1 はじめに

ここでは, 数式処理システム Mathematica を用いた数学研究について報告する. 扱うテーマは確率論的なゲームと組み合わせゲーム論である. 本文の途中にどのように使ったかを書くが, 今回の数学研究において Mathematica の機能のうちで, 重要であったポイントは3つある.

1. リスト操作の容易さ.
2. 手続き型プログラムと関数型プログラムの両方が使えること.
3. 数学の式と殆ど同じ形で関数を定義できること.

これらの特徴をフルに使うことで, 興味深いことがいくつも発見された.

なお, この数学研究には関西学院大学理工学部生の高野 凌史, 水田 和成, 廣川 快, 戸國 友貴と東北大学工学部生の中屋 悠資, 関西学院高等部の鈴木 翔大, 北川 将が参加し, 著者達と共同研究を行った.

2 実践例

2.1 ロシアンルーレットとカードゲームから生成されるパスカル的三角形

ロシアンルーレットは, ルーレットという名の通り必勝法はなく, 確率的に勝敗が決まるゲームである. このゲームには非常に美しい数学的構造があり, 著者の一人が [1], [2] において発表している. ロシアンルーレットは拳銃の中に弾を何発かこめ, 2人のプレイヤで引き金を自分に向かって交互に引いていき, 最初に弾があたったプレイヤが負けとなるゲームである.

[1] では, ロシアンルーレットというテーマがあまり教育的でないこともあり, 定義 2.3 のカードゲームとして扱ったが, この論文でも同じようにする.

この定義のゲームでは, あるプレイヤが負ける確率の分母と分子がそれぞれ独立に, パスカルの三角形のような規則性 (Pascal-like property) を持つ.

例 2.1. パスカルの三角形では, 組み合わせの数 ${}_nC_m$ を並べるが, $n = 1, 2, 3, \ldots$ としながら, 上から下へ並べて, $m = 0, 1, \ldots, n$ と左から右に並べる. 私達の研究でも同じようにする. ただし $m = 1, 2, \ldots, n$ と左から並べる. $m = 0$ では赤のカードが 0 となり, ゲームが成立しないからである.

先行研究によれば, 集合 $\{f(p, n, m, s) : m \leq n, n = 1, 2, \ldots\}$ は, 正の整数 p と $s = 1$ に対してパスカルの三角形と似たパターンを有することが示されている. 例として, カードゲームを $p = 4$ 人で行ない, 各自が自分の番になると $s = 1$ 回ずつカードを引く場合を考えて, 先手のプレイヤーの負ける確率を考える. その確率をパスカルの三角形と似たやり方で並べる. すなわち, $\{f(4, n, m, 1), 1 \leq m \leq n, n = 1, 2, \ldots, 6, 7\}$ から作られる Pascal-like triangle を図 1 に示す.

見て明らかなように, 図 1 は美しい特徴を有する. 例えば, 図 1 によれば, $f(4, 6, 2, 1) = \frac{6}{15}$, $f(4, 6, 3, 1) = \frac{10}{20}$, $f(4, 7, 3, 1) = \frac{16}{35}$, $6 + 10 = 16$, $16 + 20 = 35$ であり, 分子と分母で Pascal-like triangle が作られる. 図 2 の数字は, 図 1 にある分数の分子である.

$$
\begin{array}{c}
\frac{1}{1} \\
\frac{1}{2}, \frac{1}{1} \\
\frac{1}{3}, \frac{2}{3}, \frac{1}{1} \\
\frac{1}{4}, \frac{3}{6}, \frac{3}{4}, \frac{1}{1} \\
\frac{2}{5}, \frac{4}{10}, \frac{6}{10}, \frac{4}{5}, \frac{1}{1} \\
\frac{2}{6}, \frac{\mathbf{6}}{\mathbf{15}}, \frac{\mathbf{10}}{\mathbf{20}}, \frac{10}{15}, \frac{5}{6}, \frac{1}{1} \\
\frac{2}{7}, \frac{8}{21}, \frac{\mathbf{16}}{\mathbf{35}}, \frac{20}{35}, \frac{15}{21}, \frac{6}{7}, \frac{1}{1}
\end{array}
$$

図 1: Pascal-like triangle

$$
\begin{array}{c}
1 \\
1,1 \\
1,2,1 \\
1,3,3,1 \\
2,4,6,4,1 \\
2,6,10,10,5,1 \\
2,8,16,20,15,6,1
\end{array}
$$

図 2: 図 1 の分子を取り出したもの

イギリスの数学者がこの三角形を Pascal-like triangle と呼んだので, 私達はパスカルの三角形に似た三角形を Pascal-like triangle と呼ぶことにする.

注意 2.1. 1 番目のプレイヤーが負ける確率は, 1 番目のプレイヤーが負けるカードの並び方の数を, カードのすべての並べ方である, ${}_nC_m$ で割ることで得られる. すると確率の分母は ${}_nC_m$ となるので, 分子に来るべき 1 番目のプレイヤーが負けるカードの並び方の数がパスカル的な性質を持てば, 結果として分母と分子が同時にパスカル的性質を持つことになる.

定義 2.1. 定義 2.3 において, 先手のプレイヤーが負けるカードの組み合わせの数を $U(p, n, m, s)$ とする.

この時次の定理が成り立つ.

定理 2.1.

$$U(p, n+1, m+1, s) = U(p, n, m+1, s) + U(p, n, m, s). \tag{2.1}$$

ここで考えているゲームの確率は $f(n, m, s) = \frac{U(p,n,m,s)}{{}_nC_m}$ となるので, このゲームで先手プレイヤーが負ける確率は, 分母と分子がそれぞれパスカルの三角形のような性質を持つ.

2.1.1 二人でのカードゲームの一般化

2名のプレイヤーの場合でも, 条件を少し一般化すると, 状況は非常に複雑になる. 例えば, 引き分けがないという条件が成立すると, パスカル的性質が満たされることを定理2.2と定理2.3で得た. なお, 証明は省く.

定義 2.2. 二人のプレイヤーが交互に, それぞれ s_1, s_2 回ずつカードを引く. それぞれが g_1, g_2 回のカードを集めると負けることになる.

定理 2.2. このとき, 先手のプレイヤーが負けるような場合の数は次のようになる.

$$
\begin{aligned}
&U(n, m, s_1, s_2, g_1, g_2) = \\
&\sum_{h=1}^{s_1}\Bigg(\sum_{k=\lceil \frac{g_1-h}{s_1}\rceil+1}^{\min(\lfloor \frac{n-m+g_1-h}{s_1}\rfloor+1, \lfloor \frac{n-h}{s_1+s_2}\rfloor+1, \lfloor \frac{n-m+g_1+g_2-h-1}{s_1+s_2}\rfloor+1)} \\
&\quad \sum_{v=\max(0, m-n+(k-1)(s_1+s_2)-g_1+h)}^{\min(g_2-1, m-g_1, (k-1)s_2)} \big({}_{(k-1)s2}C_v \times \\
&{}_{h+(k-1)s1-1}C_{g1-1} \times {}_{n-(k-1)(s1+s2)-h}C_{m-v-g_1}\big)
\end{aligned} \tag{2.2}
$$

上の定理を発見するためには, 数式処理システム Mathematica を使った. まず, ループ文を使った以下の手続き型のプログラムで組み合わせを計算できるようにしておく. ここでは, $s_1 = 2, s_2 = 3 g_1 = 4, g_2 = 2$ としている.

```
St[l_List, 0] := {{}};
St[l_List, k_Integer?Positive] :=
Block[{oneless = St[l, k - 1], i, nn = Length[l]},
Apply[Join, Table[Map[(Prepend[#, l[[i]]]) &, oneless], {i, nn}]]];
s1 = 2; s2 = 3; g1 = 4; g2 = 2;
check[dd_] :=
Block[{sen, got, t, kai, los, dd2 = dd, senloss}, kai = 1; sen = 0;
got = 0; Do[t = First[dd2];
If[1 <= Mod[kai, s1 + s2] <= s1, sen = sen + t, got = got + t];
Which[sen >= g1, senloss = 1; Break[], got >= g2, senloss = 2;
Break[], kai >= Length[dd] + 1, senloss = 3; Break[]];
dd2 = Rest[dd2]; kai = kai + 1, {n, 1, Length[dd]}]; senloss];
hhh[n_, m_] := Block[{tt},
data = Select[St[{0, 1}, n], Count[#, 1] == m &];
Count[Map[check[#] &, data], 1]]
```

次に, 数学的に考えて, 定理の式を作っていきながら, 正しいと思える式ができたならば, 下のような Mathematica で関数として表す. Mathematica では ${}_nC_m$ を $Binomial[n,m]$ と書くが, それ以外では数学の式と殆ど変わらない. そのために, 定理の数式を確かめるには都合が良い. それが上の手続き型のプログラムの数値と合えば, 式が正しいということがほぼわかる. あとは, いろんな値を代入しながら, もし異なる値になると原因を探していく. そうやって得られた式を, 厳密に数学的に証明していく. このような形で Mathematica を数学研究に使う.

次の Mathematica プログラムによって組み合わせの数を三角形に並べることができる.

$$
\begin{aligned}
&\texttt{ft[n\_, m\_] :=} \\
&\quad \sum_{\texttt{h=1}}^{\texttt{s1}} \sum_{\texttt{k=Ceiling[(g1-h+s1)/s1]}}^{\texttt{Min[Floor[(n-h+s1+s2)/(s1+s2)],Floor[(n-m+g1-h+s1)/s1],Floor[(n-m+g1+g2-1-h+s1+s2)/(s1+s2)]]}} \\
&\quad \sum_{\texttt{u=Max[0,m-n-g1+(k-1) (s1+s2)+h]}}^{\texttt{Min[s2 (k-1),g2-1,m-g1]}} \texttt{(Binomial[s1 (k-1) + h-1, g1-1] *} \\
&\qquad \texttt{Binomial[s2 (k-1), u] * Binomial[n-(k-1) (s1+s2)-h, m-g1-u]);}
\end{aligned}
$$

図 3: Mathematica プログラム

```
s1 = 2; s2 = 3; g1 = 2; g2 = 3;
t = 10; ColumnForm[
 Table[Table[ft[n, m], {m, 1, n}], {n, 1, t}], Center]
```

下の三角形において, パスカル的な性質が満たされているところと, そうでないところを探し, 数値をいろいろ変えながら, 法則を探す. そのようにして定理 2.3 を得た.

```
{0}
{0, 1}
{0, 1, 1}
{0, 1, 2, 1}
{0, 1, 3, 3, 1}
{0, 3, 10, 12, 4, 1}
{0, 6, 22, 31, 16, 5, 1}
{0, 6, 28, 53, 47, 21, 6, 1}
{0, 6, 34, 81, 100, 68, 27, 7, 1}
{0, 6, 40, 115, 181, 168, 95, 34, 8, 1}
```

図 4: Mathematica プログラム

定理 2.3.

$$m \geq g_1 + g_2 \tag{2.3}$$

が成り立つときに, 次の条件が満たされる.

$$U(n,m,s_1,s_2,g_1,g_2) = U(n-1,m-1,s_1,s_2,g_1,g_2) + U(n-1,m,s_1,s_2,g_1,g_2). \tag{2.4}$$

上の定理は, ゲームの言葉を使うと引き分けがない場合には先手の人が負ける組み合わせの数はパスカル的性質を持つことを示している. しかし, 引き分けがあってもパスカル的性質を満たす場合があるようである. それを次の予想として書く. この予想も, Mathematica の計算結果によって得られたものである.

予想 2.1.

$$g_1 \leq s_1 \tag{2.5}$$

であり, 自然数 t, u に対して,

$$n = (s_1 + s_2)(t-1) + s_1 + u \tag{2.6}$$

となるとする. このとき,

$$U(n, m) = U(n-1, m-1) + U(n-1, m). \tag{2.7}$$

またさらに, 人数を一般化すると, 以下のように定義することができる.

定義 2.3. p, n, m, s を $m \leq n$ を満たす正の整数とする. 円形のテーブルの周りに p 人のプレイヤー $\Theta_1, \Theta_2, ..., \Theta_p$ を配置し, プレイヤー Θ_1 からゲームを開始する. そして順に, 同じ大きさの m 枚の赤のカードと $n-m$ 枚の白のカードを含む箱を渡していく. プレイヤーは箱を受け取ったら, s 回カードを無作為に取り, 取ったカードは箱に戻さない. したがって, 箱の中のカードの数は減少していく. ただし, 箱の中を見ることはできないとする. ここで, ラウンドという言葉を使うが, 第 t ラウンドとは, ゲームが開始してから t 回目にカードが取り出されることである. このゲームでは, プレイヤー Θ_1 が第 $1, 2, 3, ..., s$ ラウンドにカードを引く. 次に, プレイヤー Θ_2 が第 $s+1, s+2, ..., 2s$ ラウンドでカードを引く. これを次のプレイヤー Θ_3 が続けていく. こうやっている間に, 最初に赤のカードを引いたプレイヤーが負けであり, そこでゲームは終了する.

定義 2.4. 定義 2.3 において一番最初のプレイヤー Θ_1 が負ける確率を $f(p, n, m, s)$ とする.

注意 2.2. 定義 2.3 は, 数学的にはロシアンルーレットのゲームだと言われた方がわかりやすいと人が多いだろう. シリンダーの中で, 弾の入っていない所で引き金を引くことが白のカードを引くことに相当し, 弾が入っている所で引き金を引くことが赤のカードを引くことに相当する. このゲームでは p 人のプレイヤーがいるが, 簡単なため最初のプレイヤーの確率だけを扱っている. ロシアンルーレットのデータ構造では, 図 5 のような小さな正方形の集まりからなるシリンダーの中に実弾を配置しておいて, 一番左の弾倉から順に引き金を引くことになる. カードゲームの場合でも, 図 5 の中に, カードが並んでいて, 左端から取り出されると考えて計算する. なお, プレイヤーはカードがどのように並んでいるかは知らないとする.

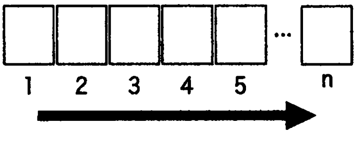

図 5: カードの並び・シリンダーの並び

このように, 生徒にとって興味を持ちやすい題材を活用し, 簡単な事例からどんどん一般化していくことで, 多くの成果をあげることができている. またこの研究に数式処理システムが大きく寄与しており, 今後様々な数式処理システムについて検討し, 多くの現場で実践できるモデルを構築していく必要があるかと考えられる.

2.2 Wythoffのゲームの変種

組み合わせゲーム論において多くの人が研究してきたWythoffのゲームの変種について研究する.

まず, 組み合わせゲーム論について説明する. ゲームの理論には大きく分けて, 2種類の分野がある. 1つは, 経済学などで多く用いられる, 偶然性が入ったゲームである. ゲーム理論と言われるときは, この分野のことである場合が多い. もう1つが組み合わせゲーム論と言われる分野で偶然性が入らないものである. 経済学的なゲーム論に比べて, 組み合わせゲーム論が本格的に発展し始めたのは計算機を使うことが容易になってからである. 偶然性が入らないゲームには, 将棋, チェス, 囲碁などがあるが, これらのゲームを考えればわかるように, あるゲームの状態が必勝位置か, そうでないかを調べるためには数百手先を読むことなどはごく普通のことであり, これは人間が普通にできる範囲を超えている. 計算機による計算が重要な手段となっている分野である.

組み合わせゲーム理論については, 文献 [9, 6, 8, 5, 4] などに詳しい.

本稿では, [3] で研究されている Corner the Queen 問題の変種である Corner the Two Rooks 問題について述べる. Corner the Queen 問題で用いられている Queen の代わりに, 我々はチェスの Rook を2つ用いる. この問題は, 駒が他の駒に乗ることを禁止しながら, 飛び越すことを許すと, 2次元マヤゲームとして考えることができる. $\mathcal{P}$-positions の集合と $\mathcal{N}$-positions の集合は Nim 和で表すことができるが, 従来の Nim とは異なる数学的な構造を持つ. いくつかの $\mathcal{P}$-positions の Nim 和は正であり, いくつかの $\mathcal{N}$-positions の Nim 和は0である. なお, ここで発表する研究は筆者達の研究 [7] の変種と考えることもできる.

2.2.1 Corner the Two Rooks 問題 2次元マヤゲーム

$Z_{\geq 0}$ を非負整数の集合とする. ここではチェスの伝統を変え, チェス盤の欄を数字の対で表す. この盤の一番左上の角を (0,0) と表し, 他の座標は右方向を x, 下方向を y とする (図6を参照のこと).

定義 2.5. (*i*) Corner the Two Rooks 問題は, 次のような問題である: 2つの Rook を下方向と右方向に無制限の大きさを持つチェス盤の上に配置し, 2人のプレイヤーが Rook のどちらか一つを選んで動かす. Rook は左方向と上方向にどこまでも好きなだけ動かすことができる. そして, Rook は別の Rook を飛び越えることができるが, 別の Rook の上に置く (あるいは別の Rook と同じ位置に動かす) ことはできない. Rook を動かすことができなくなったプレイヤーが負けとなる.
(*ii*) 2つの Rook の座標を (x,y,z,w) で表す: ここで, (x,y) は1つ目の Rook の座標, (z,w) は2つ目の Rook の座標である.
(*iii*) 他のところへ動かせなくなる座標, つまり終了状態 $\mathcal{E}$ の集合は
$\mathcal{E} = \{(0,0,1,0),(0,0,0,1),(1,0,0,0),(0,1,0,0)\}$ である.

注意 2.3. (*i*) 定義2.5より, 2つの Rook には違いはない. すなわち (a,b,c,d) は (c,d,a,b) と同じである.

(ii) 定義 2.5 より, 全ての Rook の座標 (x, y) は, x と y に関して対称である. それゆえに, 座標 (x, y, z, w) における証明は, (y, x, w, z) の証明を満たす.
(iii) 定義 2.5 より, 2 つの Rook は同じ座標に配置することができない. つまり全ての (x, y, z, w) に対して, $(x, y) \neq (z, w)$ である.

注意 2.4. 定義 2.5 において, $move((x, y, z, w))$ は, 一方の Rook が他方の Rook と同じ座標を取ることができないことから, $\{(x, y, x, y), (z, w, z, w)\}$ を含まない.

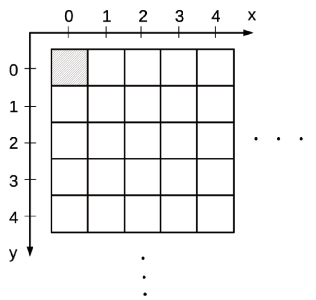

図 6: 座標の定義

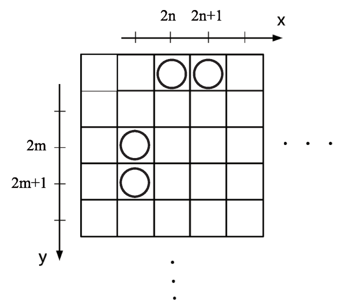

図 7: $\mathcal{P}_1$ の例

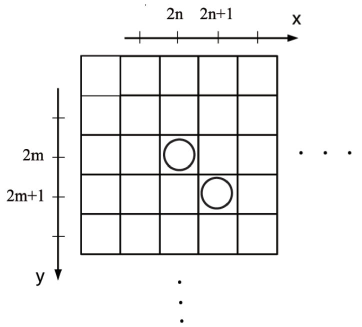

図 8: $\mathcal{N}_0$ の例

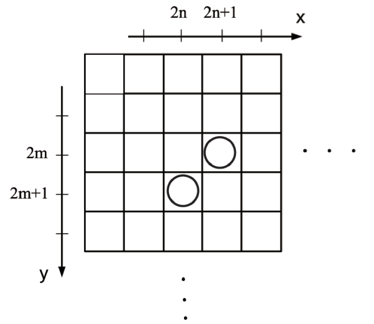

図 9: $\mathcal{N}_0$ の例

Corner the Two Rooks 問題は, 引き分けのない不偏ゲームであることから, 以下に示す 2 つの帰結類のいずれかに属する.

定義 2.6. (a) $\mathcal{N}$-position とは, 先手のプレイヤーが最適な戦略を使うことにより, 後手のプレイヤーがどのような戦略を用いても, 先手が必ず勝利できる状態のことである.
(b) $\mathcal{P}$-position は, 後手のプレイヤーが最適な戦略を使うことによって, 先手のプレイヤーがどのような戦略を使っても, 後手が必ず勝利できる状態のことである.

定義 2.7. $x, y, z \in Z_{\geq 0}$ に対し, 以下のように定義する.

$$
\begin{aligned}
\mathcal{P}_1 &= \{(2n, m, 2n+1, m) : n, m \in Z_{\geq 0}\} \cup \{(2n+1, m, 2n, m) : n, m \in Z_{\geq 0}\} \\
&\quad \cup \{(n, 2m+1, n, 2m) : n, m \in Z_{\geq 0}\} \cup \{(n, 2m, n, 2m+1) : n, m \in Z_{\geq 0}\}, \qquad (2.8) \\
\mathcal{N}_0 &= \{(2n, 2m, 2n+1, 2m+1) : n, m \in Z_{\geq 0} \\
&\quad \cup \{(2n+1, 2m+1, 2n, 2m) : n, m \in Z_{\geq 0}\} \\
&\quad \cup \{(2n+1, 2m, 2n, 2m+1) : n, m \in Z_{\geq 0}\} \\
&\quad \cup \{(2n, 2m+1, 2n+1, 2m) : n, m \in Z_{\geq 0}\}, \qquad (2.9) \\
\mathcal{P} &= (\{(x, y, z, w) : x \oplus y \oplus z \oplus w = 0 \text{ and } x, y, z, w \in Z_{\geq 0}\} \cup \mathcal{P}_1) - \mathcal{N}_0, \qquad (2.10) \\
\mathcal{N} &= \{(x, y, z, w) : x \oplus y \oplus z \oplus w \neq 0 \text{ and } x, y, z, w \in Z_{\geq 0}\} \cup \mathcal{N}_0 - \mathcal{P}_1. \qquad (2.11)
\end{aligned}
$$

例 2.2. 集合 $\mathcal{P}_1$ に属する要素の例は図 7, 集合 $\mathcal{N}_0$ に属する要素の例は図 8, 図 9 を参照のこと.

定理 2.4. $\mathcal{P}$ と $\mathcal{N}$ は, それぞれ $\mathcal{P}$-positions と $\mathcal{N}$-positions の集合である.

下の Mathematica プログラムが 2 次元 Maya ゲームの $\mathcal{P}$-position を計算するためのもので, 座標は 0 から 10 までを扱っている. そして, 得られた $\mathcal{P}$-position の座標の集合について, 排他的論理和 (ニム和) を計算すると, 下の 0 と 1 の集合が現れる. このような計算結果を基にして, 予想を作り, 定理を作っていく.

```
ss = 10; al =
Flatten[Table[{a, b, c, d}, {a, 0, 4}, {b, 0, ss}, {c, 0, 4}, {d, 0, ss}], 3];
allcases = Select[al, ! ({#[[1]], #[[2]]} == {#[[3]], #[[4]]}) &];
move[z_] := Block[{p}, p = z;
Union[
Complement[
Table[{t1, p[[2]], p[[3]], p[[4]]}, {t1, 0,
p[[1]] - 1}], {{p[[3]], p[[4]], p[[3]], p[[4]]}}],
Complement[
Table[{p[[1]], t2, p[[3]], p[[4]]}, {t2, 0,
p[[2]] - 1}], {{p[[3]], p[[4]], p[[3]], p[[4]]}}],
Complement[
Table[{p[[1]], p[[2]], t3, p[[4]]}, {t3, 0,
p[[3]] - 1}], {{p[[1]], p[[2]], p[[1]], p[[2]]}}],
Complement[
Table[{p[[1]], p[[2]], p[[3]], t4}, {t4, 0,
p[[4]] - 1}], {{p[[1]], p[[2]], p[[1]], p[[2]]}}]]];
Mex[L_] := Min[Complement[Range[0, Length[L]], L]];
Gr[pos_] := Gr[pos] = Mex[Map[Gr, move[pos]]];
pposition = Select[allcases, Gr[#] == 0 &];
Map[BitXor[#[[1]], #[[2]], #[[3]], #[[4]]] &, pposition]
```

```
{1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,
0,1,1,0,1,1,0,1,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,
0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,1,1,0,1,0,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,
1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,1,1,0,1,1,0,0,1,0,0,1,1,0,0,
```

```
0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,0,1,1,0,0,1,1,0,
1,1,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,
0,1,0,0,0,0,1,1,1}
```

この問題も, 生徒にとって興味を持ちやすい題材であり, 簡単な事例からどんどん一般化していくことで, 多くの成果をあげることができている. これらの内容についてさらに研究を進めていき, 生徒の活躍の場を広げていくことが重要であると考えている.

3 まとめ・今後の展望

本稿では,Mathematica を用いて, 生徒や学生が組み合わせゲームや確率論的なゲームを題材とした数学研究によって得られた成果について報告した. これらの結果は先進的な事例であり, 多くの現場では到底できないと考えられているが, 広めていく活動を展開していくことが求められる. それに際し, 筆者の一人は創造性育成をプログラミング教育やテクノロジーを用いて実現するという観点から研究を進めており [12, 18, 17, 15, 16], 創造的態度が高まったという結果も得られている [14, 13]. また, アクティブラーニングを指向した数学アプリも開発 [19, 20, 21] しており, そのアプリによる実践も予定している. このような知見をいかし, 一部の生徒のためのものではなく, 多くの生徒にとって有益な教育モデルの構築を行うために, ソフトウェアの選択や題材設定などを含めて, より教育的な観点から研究を進めていく必要があろう.

謝辞

本研究は, 京都大学数理解析研究所共同事業「数学ソフトウェアとその効果的教育利用に関する研究」による成果である.

参考文献

[1] Matsui, H., Minematsu, D., Yamauchi, T. and Miyadera, R., “Pascal-like triangles and Fibonacci-like sequences”, *Mathematical Gazette*,Volume **94**, Issue 529, p.27–p41, 2010.

[2] Matsui, H., Saita, N., Kawata, K., Sakurama Y. and Miyadera, R., “Elementary Problems”, B–1019, *Fibonacci Quarterly*, series **44.3**, p277, 2006.

[3] Wythoff,W.A., “A Modification of the Game of Nim.”, *Nieuw Arch*,Wisk.**7**, pp.199–202, 1907/1909.

[4] A.N.Siegel: “Combinatorial Game Theory (Graduate Studies in Mathematics)”, American Mathematical Society, 2013.

[5] 佐藤 文広, “石取りゲームの数学: ゲームと代数の不思議な関係”, 数学書房, pp.132–145, 2014.

[6] 一松 信, “石とりゲームの数理 POD 版 (数学ライブラリー 教養篇)”, 森北出版, pp.93–104, 2003.

[7] 宮寺 良平, 福井 昌則, 中屋 悠資, 戸國 友貴: “A Generalized Ryuoh-Nim - A Variant of the classical game of Wythoff Nim, 第 36 回 GI・第 41 回 EC 合同研究発表会, 2016.

[8] 山崎 洋平: “組み合わせゲームの裏表”, シュプリンガー・フェアラーク東京, 1989

[9] M.H.Albert,R.J.Nowakowski,D.Wolfe(川辺 治之 訳), “組合せゲーム理論入門-勝利の方程式-”, 共立出版, 2011.

[10] R.Miyadera, Y.Sakamoto, K.Mizuta, R.Takano, K.Hirokawa and M.Fukui.: “Two-Dimensional Maya Game and Two-Dimensional Silver Dollar Game”, *The 20th Japan Conference on Discrete and Computational Geometry, Graphs, and Games (JCDCG^3 2017)*, 2017.

[11] R.Miyadera, M.Fukui, M.Kitagawa, S.Suzuki, Y.Tokuni and Y.Nakaya: “Pascal-Like Triangles and Fibonacci-Like Sequences”, *The 20th Japan Conference on Discrete and Computational Geometry, Graphs, and Games (JCDCG^3 2017)*, 2017.

[12] 福井 昌則, “数学的ゲーム・パズルを用いた研究・学習活動を行うための環境構築を目的とした iOS アプリケーション開発”, *The 22nd Game Programming Workshop 2017*, pp.119–125. 2017.

[13] 福井 昌則, 森山 潤, “創造的態度の育成を目的とした高校生対象の Java プログラミング教育の実践”, 第 33 回 日本教育工学会全国大会, 2017.

[14] 福井 昌則, “創造的態度の育成を目的とする数学的ゲーム・パズルを題材としたプログラミング教育の実践”, 第 15 回 ゲーム学会 全国大会, 2017.

[15] 福井 昌則, 番庄 智也, 萩倉 丈, 森山 潤, “ブロックプログラミングライクに HTML の編集を可能とす学習用 GUI コーディング環境の開発”, 日本産業技術教育学会 近畿支部 第 34 回研究発表会, 2017.

[16] 福井 昌則, 宮寺 良平, “高校生による数学研究を支援する数式処理システムの活用とその教育”, 数式処理の新たな発展ーその最新研究と他分野との連携ー 京都大学 RIMS 研究集会, 2017.

[17] 福井 昌則, 萩倉 丈, 番庄 智也, 森山 潤, 平嶋 宗, “各教科内で Computational Thinking を育成することを目的とした Block 型言語環境の開発”, 教育システム情報学会中国支部 2017 年度研究発表会講演論文集, **17**(1), pp.17–20, 2017.

[18] 福井 昌則, “組合せゲーム理論の教育利用の可能性について”, 第 1 回日本組合せゲーム理論研究集会, 2017.

[19] 福井 昌則, “アクティブ・ラーニングを促進する数理的パズル「碁石拾い」を題材としたiPadアプリケーションの開発”. コンピュータ利用教育学会『コンピュータ&エデュケーション』. vol.**42**, pp.55–58, 2017.

[20] 福井 昌則, “数列に対する苦手意識の軽減を目的としたアクティブ・ラーニングを促進するiOS アプリケーション「ヨセフス問題」の開発”, コンピュータ利用教育学会『コンピュータ&エデュケーション』. vol.**43**, pp.85–88, 2017.

[21] 福井 昌則, 萩倉 丈, 番庄 智也, “順列組合せの単元におけるアクティブ・ラーニングを促進する iOS アプリケーション「ナイトツアー for Education」”, コンピュータ利用教育学会『コンピュータ&エデュケーション』. vol.**43**, pp.89–92, 2017.