Optimal strategy in the childrens game Memory

Erik Alfthan
eral02@kth.se

May 21, 2007

Preface

This paper was written as a master thesis in discrete mathematics at the Mathematics Department of the Royal Institute of Technology. I was tutored and examined by professor Svante Linusson within the Combinatorics Group.

Abstract

Two mathematical games are contructed from the children's game memory. One game, named Terminating memory is constructed as a two player game with rules as close to the children's game as possible. The most significant change is made in order to make the game terminate. It turns out that there are non-trivial elements of strategy in Terminating memory. Depending on the expected number of turn overs, i.e. the number of times the lead is lost, the strategy seems to be to try to force the opponent to reach a known losing position which is when the last turn over occurs. However, this could not be proven generally, but is computed for all games with less than 200 pairs.
A second game of memory that complies with the rules of combinatorial games was therefore contructed, in order to determine which elements are important to the previous game, Terminating memory. This game, Combinatorial memory was generally solved as game equivalent to a sequence of weighted misère nim games. A hypothesis of implications of this to Terminating memory was presented. It is suggested that the strategy will depend on whether there are expected to be odd or even number of nim games left, of which the last game is probably the largest. Both player are trying to reach a position where they will get the last collect sequence. This is consistant with the main conjecture of terminating memory. A general way to compute whether odd or even number of remaining nim games is most likely is needed to make this result useful.

Chapter 1

Introduction

1.1 True memory

Memory game is a children's game, where two or more players collect pairs of cards from a commonly shared board of hidden cards. All cards consist of matching pairs. In every turn, each player turns two cards face up. If they match, the player may collect the pair and make another try. If the two cards fail to make a pair, the player ends his turn. The game ends when all pairs have been collected. Neither the objective nor the exact rules are generally stated in a clear mathematical way. Exact rules of when a player turn is end or how many players there are etc. must be stated. The original game was therefore translated into two different games.

1.2 Terminating memory

In chapter 2 a game called terminating memory is constructed. This game tries to be as close to true memory as possible. Most of the limitations are interpretations of the children game. The objective for constructing this game is to determine an optimal strategy when playing the game described in the above section, the true memory game. However some move that are allowed in true memory are disallowed in terminating memory in order to make the game end. No such condition is imposed on the players in true memory.
Also the players of terminating memory are slightly different from the typical five year old player of true memory. Both players are assumed to have perfect memory and try to collect as many pairs as possible, over the entire

game, rather than the slightly more irrational moves of a child that often but not always try to collect as many pairs as possible in the same turn. The general recursion is stated in section 2.2. In section 2.3 some examples and initial results are described. In section 2.4, computed results shows that there is a non-trivial element of strategy in terminating memory, as the computed strategy table (Table 2.1) shows, the strategy depends on the parity of remaining unknown cards. The optimal strategy of the game could not be generally determined, even though the rather simple pattern seemed to exist. A conjecture that strategy will depend only on the number of turn-overs left in the game is proposed in section 2.5.

1.3 Combinatorial memory

In this section, tools of determine strategy of terminating memory is developed by changing the game into an entirely combinatorial game, as described in for example Winning ways, 2nd ed., Berlekamp et al. 2001. These games are simpler to analyse but may generate insight of what the outcome of the general game depends on. In section 3.1, some general background to combinatorial games is given, followed in section 3.2 by a special case of nim that occurs in the constructed game. Section 3.3 displays some examples of how a board of memory should be interpreted as combinatorial memory. In the following sections, combinatorial memory is generally solved, any game of combinatorial memory is game equivalent to a sequence of misère nim games, weighted with the number of pairs that may be collected after every nim game. Even though this game could be generally solved, the insights learned generated tools too heavy to be used to generally solve terminating memory within the scope of this paper. However, a hypothesis could be put forward, stating that if the expected number of remaining nim games could be calculated, the conjectures of terminating memory could be verified.

Chapter 2

Terminating memory

2.1 Introduction

The original children's game of memory, as described in section 1.1, does not uniquely define a computable game. In this chapter, a computable variant of memory is defined and analysed. It is called terminating memory since the most significant limitations is made in order to have determined maximum number of turns before it ends.
Other changes are that a fixed number of players, two, is used and that both players will have perfect memory, i.e. any card that is revealed in a turn is henceforth considered known to both players. Both players are assumed to play optimal, considering the known information, and with the same objective.
There are two possible objectives, to maximise the probability to collect more pairs than the opponent, or to maximise the expected number of pairs collected. In many cases, the objectives will coincide. In order to be able to disregard game history, the object in terminating memory will assume that the target is to maximise the expected number of pairs collected.

2.1.1 Termination

In order to determine what is needed to make every possible board of memory to terminate, one first have to conclude when a game memory ends. Obviously, this is when there are no more pairs on the board. Assuming that both players play according to the objective to collect as many pairs as possible, any known two cards that make a pair will be collected. Thus, the limitation needed must force the game not to enter a stalemate where no unknown card is opened. In an impartial game, that is when both players are allowed to do the same moves, no moves where a player ends his turn in the same position as he started in is allowed. Both players will be allowed to make the same moves, given a game situation. If any of the players want to make a move that does not open an unknown cards; which does not change the board, since no pair can be collected and no unknown card is opened; neither player will change the board and the game will not end. The allowed moves will be stated in section 2.1.3.

2.1.2 Optimal playing

Not only the objective, but also the premises of how the best way to reach the objective is calculated must be clear. Both players are assumed to play optimal playing, that is, they will make the best allowed move to maximise the expected number of pairs collected under the assumption that the opponent may do the same. Neither player will make any mistake, thus making any move only in order to invite a mistake non-optimal. Optimal play returns the best possible result against any opponent, it benefits from mistakes but uses no tactics to invite to them.

2.1.3 Definitions

In order to be able to describe the game in a uniform way, some definitions are needed.

Describing variables. In every state of memory, the information needed is given by a description of the board. The board can be described by any two variables that give the size of the board and the relation of known and unknown cards. Due to game mechanics as described in section 2.3, size of the board is given by the number of pairs on the board, the amount of known information is somewhat contra intuitively represented by a variable called *unknown pairs*. An unknown pair is a pair of which none of the cards are known. In the beginning, all pairs are unknown. When an unknown card is opened it either matches a previously opened card, leaving one pair less on the board, or fails to match. Then the opened card belonged to an unknown pair that is now broken, leaving the same number of pair on the board, but one less unknown pair.

Position (n, j) is describing a state of the board, n is the number of pairs on the board, j is the number of unknown pairs.

Expected number of pairs, E_n^j is the number of pairs the player that begins his turn in the position (n, j) is expected to collect through the remaining game under the assumtion that both players are playing optimal play.

Expected share of the pairs, $\frac{E_n^j}{n}$ is the same as E_n^j, but related to the size of the board.

Left and Right player. In order to efficiently describe a game sequence, the starting player is named Left and the other player is named Right.

Match: reveal an unknown card that makes a pair together with a previously known card.

Miss: reveal an unknown card that does not make a pair with any previously known card.

Optimal player

The expected number of pairs for Left to collect, given a position, is the number of pairs Left will collect in this turn, plus the number of pairs remaining at the end of Lefts turn, minus the number of pairs Right is expected to collect given the position when Lefts turn is ended. Since Left and Right play are assumed to play in the exact same manor, the expected value function of Left and Right in the same position, is the same. Thus, the expected number of pairs Left can expect to collect given a starting position can be recursively computed, as soon as any position that can be reached from the starting position is computed.

Strategies

In order to be able to calculate the best possible move in any position, the allowed moves must be stated. Since the choice of unknown card is irrelevant, the player can only choose whether to open a known card or an unknown card. When the first card is opened, the Optimal player is not allowed to choose the second card if it is possible to match with a known card. This leaves the following strategies,

Bad: open a known card, then an unknown that may match,

Safe: open an unknown card, then a known card, matching if possible,

Risky: open an unknown card, then match if possible, i.e. an known matching card if possible, otherwise an unknown card that may match, and

Passive open two, known, unmatching cards.

Bad is dominated by safe. Without the information of the revealed unknown card, the choice of the known card is random instead of chosen as matching or unmatching. If the revealed unknown card matches a known card, the pair may be lost if the Bad strategy is used but not if the Safe strategy is used. Otherwise, Bad and Safe strategy will have the same outcome. Therefore, the Bad strategy is henceforth disregarded.
Safe is not dominated by any other strategy and also fulfils the termination condition[1], at least one unknown card is revealed.
Risky is not dominated by any other strategy and also fulfils the termination condition, at least one unknown card is revealed.
Passive is not dominated by any other strategy, it is actually quite easy to determine when it should be used. When the expected share of the pairs with Safe or Risky strategy is less than half, the Passive strategy will be the best strategy, since no move by Left from the position is better for the Left than the Right.
However, Passive strategy does not reveal any information and thus does not fulfil the termination condition. In fact since terminating memory is impartial; that is, given a position, both Left and Right are allowed to make the same moves; and the Passive move does not alter position, any Passive move by Left will be followed by Passive move by Right. This also confirms the need of the termination condition. The **Passive** strategy will hence be disallowed.
The element of strategy disappears when no choice is possible or needed. No choice is possible when there are either no unknown cards or no known

[1]Defined in section 2.1.1

cards. When there are no unknown cards left, the game is ended, thus can not occur. However, in any position where the last known pair is matched, or when the game starts, there will be no known cards to choose. Thus, in any such position, the player in turn must choose two unknown cards.

2.2 Recursion

As stated in section 2.1.3, the expected value function E_n^j can be recursively calculated. Since Left player will chose the optimal strategy, the recursion can be stated as

$$E_n^j = \max\left({}^{safe}E_n^j, {}^{risky}E_n^j\right),$$

where ${}^{safe}E_n^j$ is the expected number of collected pairs if the Safe strategy is used and ${}^{risky}E_n^j$ is the expected number of collected pairs if the Risky strategy is used. Thus, there are two different recursions depending on choice of strategy.

2.2.1 Risky strategy

In terms of miss and match, there are three possible outcomes. If the first card is matching, the second card is automatically chosen, if the first card misses, the second choice of card may match or miss. However if the second card is matching, it may match either a previously known card, or the first opened card. Thus there are four possible outcomes of a risky move, summing together to

$$\begin{aligned} {}^{risky}E_n^j &= \frac{4j(j-1)}{(n+j)(n+j-1)}(n - E_n^{j-2}) + \frac{2j}{(n+j)(n+j-1)}(1 + E_{n-1}^{j-1}) + \\ &+ \frac{2j(n-j)}{(n+j)(n+j-1)}(n - (1 + E_{n-1}^{j-1})) + \frac{n-j}{n+j}(1 + E_{n-1}^j). \end{aligned} \tag{2.1}$$

Miss both cards

If Left choose to make the risky move and both cards miss, Right will start his turn in the position $(n, j-2)$. The probability of this is the probability to first chose one of the $2j$ cards belonging to an unknown pair, then one of the $2(j-1)$ cards belonging to remaining unknown pairs, out of the possible $n+j$ cards and $n+j-1$ cards, respectively. No pairs are collected by either player. Left can expect to collect the pairs Right does not from position $(n, j-2)$. This results in the following addition to the expected value of the Risky strategy,

$$\frac{4j(j-1)}{(n+j)(n+j-1)}(n - E_n^{j-2}).$$

Miss first card, match second

If the first card misses, but the second match with the first, Left player may collect the pair and make another move, now with one pair and one unknown pair less. The number of cards among the $n + j - 1$ to chose from to match the first, is only one. This results in the following addition to the expected value of the Risky strategy,

$$\frac{2j}{(n+j)(n+j-1)}(1 + E_{n-1}^{j-1}).$$

Miss first card, match second badly

More likely than to match the first opened card however, is to match any of the $n-j$ previously known cards. The Right player will then collect the now known pair and start the next move. This results in the following addition to the expected value of the Risky strategy,

$$\frac{2j(n-j)}{(n+j)(n+j-1)}(n - (1 + E_{n-1}^{j-1})).$$

Match first card

The last possible outcome is that the first card actually match one of the $n - j$ known cards. Then Left player receive a pair and may move again. This results in the following addition to the expected value of the Risky strategy,

$$\frac{n-j}{n+j}(1 + E_{n-1}^{j}).$$

Sum of possible outcomes from a Risky strategy

There are no other possible outcomes of a Risky strategy, the expected number of pairs will be the sum of the terms stated in sections 2.2.1 - 2.2.1,

$$\begin{aligned} {}^{risky}E_n^j &= \frac{4j(j-1)}{(n+j)(n+j-1)}(n - E_n^{j-2}) + \frac{2j}{(n+j)(n+j-1)}(1 + E_{n-1}^{j-1}) + \\ &+ \frac{2j(n-j)}{(n+j)(n+j-1)}(n - (1 + E_{n-1}^{j-1})) + \frac{n-j}{n+j}(1 + E_{n-1}^{j}). \end{aligned}$$

2.2.2 Safe strategy

When the Safe strategy is used, there is only one random event, since only one unknown card is open, no matter what it turns out to be. If it matches a previously known card, Left may collect a pair and make new move. Otherwise a non-matching, known card is chosen and Right will make a move. This results in the following recursion

$$^{safe}E_n^j = \frac{2j}{n+j}(n - E_n^{j-1}) + \frac{n-j}{n+j}(1 + E_{n-1}^j). \tag{2.2}$$

Miss first card

In order to miss the first card, the opened card must be one of the $2j$ cards initially belonging to an unknown pair. Then Right will start next turn, with one less unknown pair on the board, giving the following addition to the expected value function

$$\frac{2j}{n+j}(n - E_n^{j-1}).$$

Match first card

Otherwise, Left will match a pair consisting of one of the known cards and the first card. Left will then collect the pair and make another move, with a board consisting of one less pair, giving the following addition to the expected value function,

$$\frac{n-j}{n+j}(1 + E_{n-1}^j).$$

Sum of possible outcomes from a safe strategy

Summing the two possible outcomes into the expected value function if the Safe strategy is used, results in the following expression,

$$^{safe}E_n^j = \frac{2j}{n+j}(n - E_n^{j-1}) + \frac{n-j}{n+j}(1 + E_{n-1}^j).$$

The total of E_n^j is a maximum of a two variable recursion formula with non-constant coefficient. There is no sure general way of concluding an explicit expression.

2.3 Examples

The strategic value of different moves in memory might not be obvious, therefore some small examples are given to display some game mechanics.

2.3.1 Four pairs, $n = 4$

No unrevealed pair, $j = 0$

Assume Left player starts his turn in the position $(4, 0)$. Then the board will be as figure 2.1 describes. Since there are no unknown pairs, all four

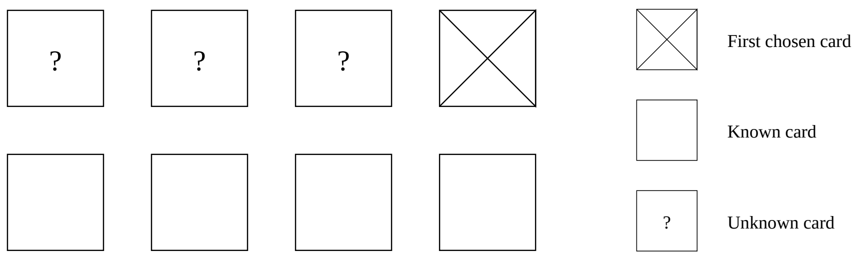

Figure 2.1: Picture of a board with 4 pairs with no unrevealed pairs

of the unknown cards will match a known card. Since both allowed move Left can do includes opening an unknown card, any move Left may choose will result in first choosing a unknown, matching card and move again. As the recursion states, Left will collect a pair and move again in the position $(3, 0)$, from which all allowed moves also have determined outcome, all *three* unknown cards will match a known. Thus, if Left starts his turn in the position $(4, 0)$, he will collect all four pairs in his turn, $E_4^0 = 4$.

One unknown pair, $j = 1$

Assume Left player start in the position $(4, 1)$, as shown in figure 2.2. The position $(4, 1)$ is slightly more difficult to evaluate, since the first card may both miss and match. If the first card match, however since there are no difference in strategy when the first card match; both by Safe and Risky strategy, Left player claims the pair; assume that the first card miss. Then the last unknown pair is broken. If Left play by the Safe strategy, Right will start his turn in the position $(4, 0)$ and collect all pairs. If Left instead choose

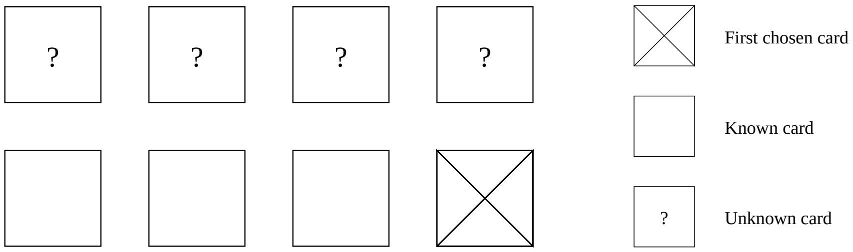

Figure 2.2: Picture of a board with 4 pairs with one unrevealed pair

the Risky strategy, the probability is $\frac{1}{4}$ that the first opened card and second opened card match. Then Left collect a pair and make his next move in the position $(3,0)$ and, by the same argument as in section 2.3.1, will collect the remaining three pairs. Thus, instead of playing the Safe strategy and lose all the remaining pair, Left can play the Risky move, with an expected number of collected pair of 1. Thus Left will make the Risky move. As shown in section 2.3.2, the Safe move is never useful when the last unknown pair is revealed. The exact value of E_4^1 can then be computed by recursion, $E_4^1 = 2$

Two unknown pairs, $j = 2$

The last example in this section is when Left starts in the position $(4,2)$. As in section 2.3.1, the difference in the Safe and Risky strategies occurs only when the first card misses. The analysis is more and more dependant of the value of the expected value functions, defined in section 2.2. Already when the board is this small, at least an estimation of the position $(3,1)$ compared to $(4,1)$ and $(4,0)$ is needed to determine strategy.

The best strategy can be determined by computing the difference between the choices,

$$\begin{aligned}
{}^{safe}E_4^2 - {}^{risky}E_4^2 &= \frac{4}{6}\Big((4 - E_4^1) - \frac{1}{5}(2(4 - E_4^0) + (1 + E_3^1) + 2(4 - 1 - E_3^1))\Big) \\
&= \frac{4}{6}\Big(4 - E_4^1 - \frac{7 - E_3^1}{5}\Big) \\
&= \frac{4}{30}\Big(13 - 5E_4^1 + E_3^1\Big) \\
&= \frac{4}{30}\Big(3 + E_3^1\Big) > 0
\end{aligned}$$

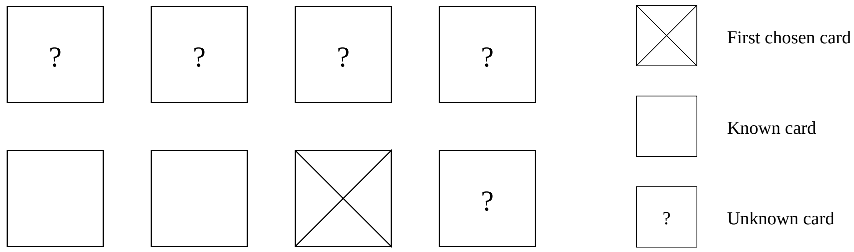

Figure 2.3: Picture of a board with 4 pairs with two unrevealed pairs

Thus the strategy should be safe. However both $E_4^0 = 4$ and $E_4^1 = 2$ were used to determine strategy. The need to determine strategy within the recursion is a problem that must be solved, by explicit computation of the recursion or by some adequate estimations of E_n^j. The computed results show that the function is not very smooth, making this very difficult to do in general.

2.3.2 Arbitrary many pairs, n free

No unknown pair, $j = 0$

When there are no unknown pairs, any opened first card must match one of the known cards. This follows from a simple proof of counting. There are no other unknown cards left, except cards matching the known cards. If there were some other unknown card, it must be belonging to an unknown pair, but there are none. Thus, independent of strategy, if Left starts in any position $(n, 0)$, Left will collect all the n remaining pairs,

$$E_n^0 = n.$$

Proof. If $n = 1$, Left will collect the pair, by opening the two remaining cards. Assume the statement to be true for some n.
Then, by induction it follows,

$$E_{n+1}^0 = 1 + E_n^0 = n + 1,$$

since from any position $(n + 1, 0)$, Left will collect the next pair and make another move. $\square$

One unknown pair, $j = 1$

The calculation of E_n^0 renders it possible to recursively compute the functions E_n^j for larger, fixed j by standard methods of solving recursions with rational coefficients in one variable, which can be found in for example Enumerative Combinatorics I, chapter 4 [2], but only when the strategy is known and equal for all n involved in the recursive calculation.

Proposition The optimal strategy in any position $(n, 1)$ is risky strategy.

Proof. If $j = 1$ when Left starts his turn, he will have to make a choice of strategy only when the last unknown pair is broken. If he uses the Safe strategy, Right will collect the all remaining pairs. If Left chooses the Risky strategy, he may match his first card and collect at least one more pair. Actually, if Left collects the last unknown pair, Left will collect all the remaining pair. Nonetheless, if the Risky strategy is used, Left has a possibility to collect at least one pair. If the Safe strategy is used, Left will not collect any more pairs. □

The initial values and the recursion is needed in order to calculate the explicit function.

$$
\begin{aligned}
E_1^1 &= 1 \\
E_n^1 &= \frac{2}{n+1}\frac{n}{n} + \frac{n-1}{n+1}E_{n-1}^1
\end{aligned}
$$

The resulting function is

$$E_n^1 = \frac{2+n}{3}.$$

Two unknown pairs, $j = 2$

When both E_n^0 and E_n^1 is calculated, all parts of the recursion in section 2.2 makes sense. Thus, as soon as the strategy can be determined, the explicit expression of E_n^2 can be computed by the same methods as in section 2.3.2. No initial values are needed since only positions where E_n^j is already computed can be reached.

Proposition The optimal strategy in any position $(n, 2)$ is safe strategy.

Proof. By the same initial expression as in section 2.3.1, the strategy decision is made by deciding the sign of

$$(n - E_n^1) - \frac{2 \cdot (n - E_n^0) + E_{n-1}^1 + 1 + (n-3)(n - (E_{n-1}^1 + 1))}{n+1}$$

Then insert the functions E_n^0 and E_n^1 already computed,

$$\frac{2(n+1)(n-1) + 0 + 2 + (n-1) + 3 + (n-3)(2(n-2) - 3)}{3(n+1)} =$$

$$\frac{4n^2 - 14n + 24}{3(n+1)} > 0 \quad \forall n \geq 2$$

Thus the strategy is Safe, for all possible n, since at least 2 pairs are needed in order to have 2 unknown pairs. □

The initial value is given by the only possible recursion from $(2, 2)$ would be used, which is the same as if Risky strategy was chosen, since there is no known cards to choose. The resulting function is

$$E_n^2 = \frac{11n - 12}{15}$$

More unknown pairs,

By these methods, every value of E_n^j can be straightforward computed, but it also shows that the maximum function in the recursion need attention in every step in j. One solution to this problem is most likely some induction where both strategy and expected value function can be generally determined in every step. This can probably be done since there seems to be a very clear pattern of strategy, as shown in the next section. However, it was found to be beyond the scope of this paper. The function E_n^j are given in the Appendix for all n, $j \leq 16$. The behaviour of E_n^j is monotonous in n for fixed j when $j > 17$ which might be important when determining general strategy.

2.4 Computed results

The solution of both the strategy function and the expected value functions can be recursively computed. In section 2.4.1, a table of the strategy function is given and in section 2.4.2, the generated expected value functions, given the calculated strategies, is stated. In the last part of this section, a simulation is done where a player playing according to the calculated strategy function faces several simple minded opponents.

2.4.1 Strategy for states with at most 20 pairs

In table 2.1, the strategy for any position with at most 20 pairs is given. The chosen strategy is given a label $\{1, 0\}$, where 1 states that the Risky strategy should be used and 0 that Safe strategy should be used. If # is used instead, it states that either no choice of strategy was possible, since there were no known cards to choose from, or that no choice was needed, since both strategies have the same outcome.

Strategy table for optimal play
N = number of pairs on board
J = number of unknown pairs
Chosen strategy, 1 : risky, 0 : safe, # : no choice

```
 J 0 1 2 3 4 5 6 7 8 9 ...
N
 2 # 1 #
 3 # 1 0 #
 4 # 1 0 1 #
 5 # 1 0 1 0 #
 6 # 1 0 1 0 0 #
 7 # 1 0 1 0 0 1 #
 8 # 1 0 1 0 0 1 0 #
 9 # 1 0 1 0 0 0 0 0 #
10 # 1 0 1 0 1 0 1 0 1 #
11 # 1 0 1 0 1 0 1 0 1 0 #
12 # 1 0 1 0 1 0 0 0 1 0 1 #
13 # 1 0 1 0 1 0 0 0 1 0 1 0 #
14 # 1 0 1 0 1 0 0 0 1 0 1 0 1 #
15 # 1 0 1 0 1 0 0 0 1 0 1 0 1 0 #
16 # 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 #
17 # 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 #
18 # 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 #
19 # 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 #
20 # 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 #
```

Table 2.1: The strategy, except for smaller n and j, seems to be to choose the Risky strategy when j is odd and the Safe when j is even.

The pattern of the table continues, computation with as many as 200 pairs, renders that, except for very small boards, the strategy should be Safe when j is even and Risky when j is odd.

Conjecture 1 Except for smaller boards, $n < 16$, the optimal strategy is to use the risky strategy when j is odd and safe strategy when j is even.
Conjecture 2 Except for smaller j, $j < 18$, $\frac{E_n^j}{n}$ is monotone in n for fixed j, growing if j is even and declining if j is odd.

The second conjecture is probably needed in order to generally determine choice of strategy, thus establishing the actual recursion in every step.

2.4.2 Expected share of pairs for states with at most 20 pairs

In this section, the calculated expected value functions are plotted. Instead of E_n^j, for fixed j, $\frac{E_n^j}{n}$ is plotted, in order to show favourability of a certain position (n, j). The general behaviour is that after some certain n, every function with odd j is declining and every function when j is even is growing. For the displayed even functions the largest needed such n is 8, for the odd functions the largest needed n is 18. As stated in the second conjecture of section 2.4.1 this can be used in order to prove the generality of the first conjecture of the same section.

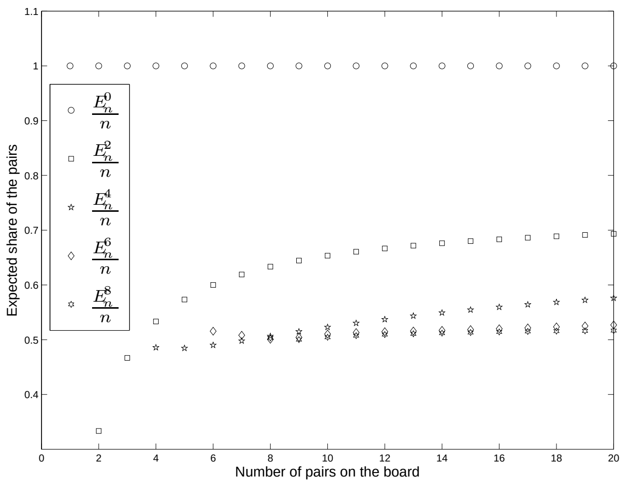

Figure 2.4: Except for $j = 4$, all even expected share functions are growing for all n. For larger n, all functions are growing.

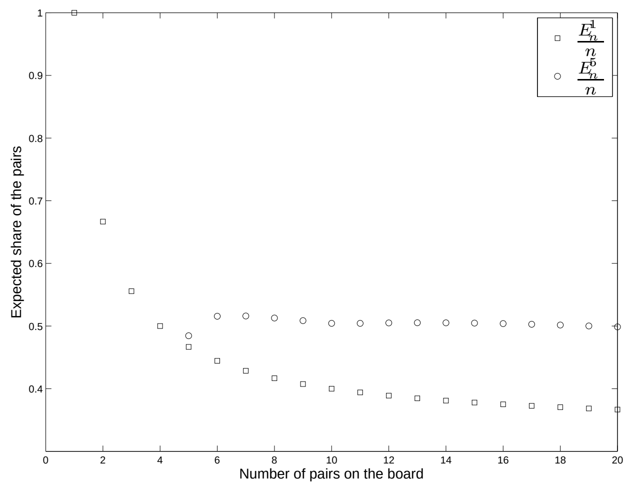

Figure 2.5: The share when $j \bmod 4 = 1$ seems to be declining from some rather small n

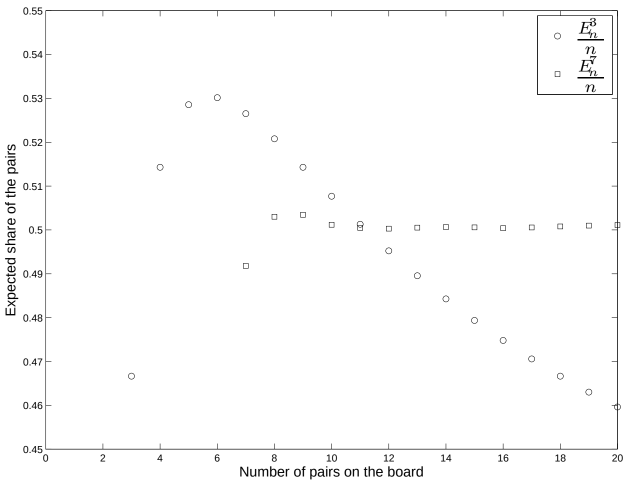

Figure 2.6: The expected share for $j = 3$ seems to be neither declining nor growing in any very predictable way.

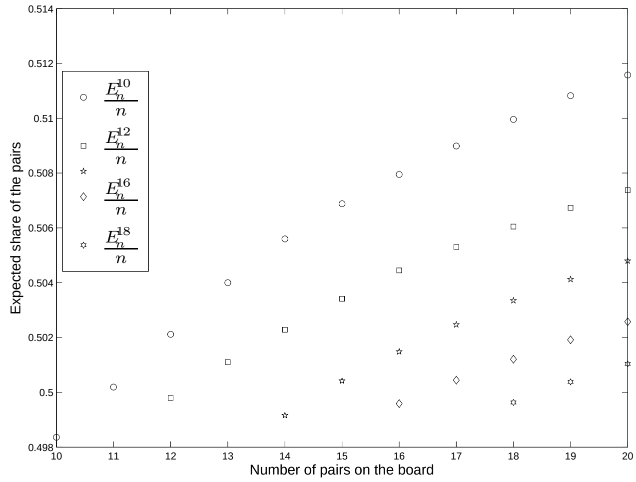

Figure 2.7: The trend of growing expected share functions seems to be more dominant when even j grows larger

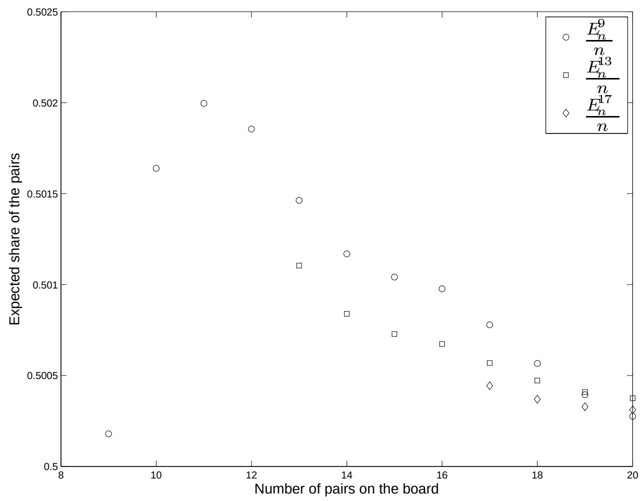

Figure 2.8: The trend when odd j mod $4 = 1$ is also more dominant for larger j, all displayed $\frac{E_n^j}{n}$ for such j are monotonously declining when $n \geq 11$

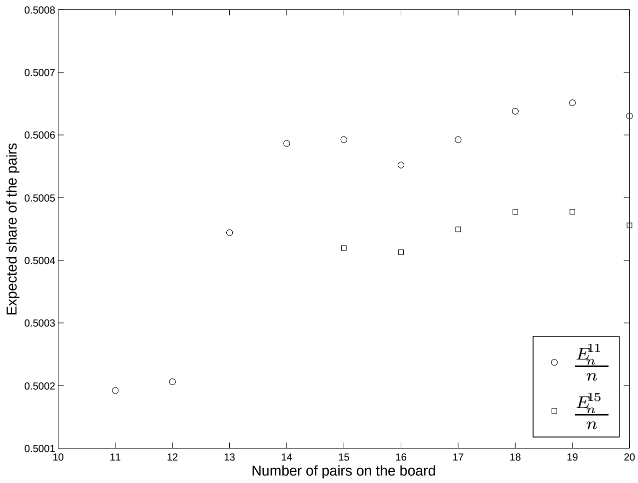

Figure 2.9: The general behaviour for the remaining $\frac{E_n^j}{n}$ is still not very clear. However, if $n \geq 19$ all computed expected share functions where odd $j \bmod 4 \neq 1$ are declining

None of the trends could be generally proven, since they all interdepend and no general way of determine strategy was found. However all the trends are verified by computer aided computation, the conjectures of section 2.2 are true for all computed values of n, j, made up to 200 pairs.

2.4.3 Simulated games versus other opponents

It is now established what is optimal play if two Optimal players play each other, for boards up to 200 pairs. If this Optimal player was to face other opponents, how valuable is the strategy?
In order to answer to this, a simulation, where the Optimal player faces different opponents; with simple and easy to describe strategies; was made. The simulations were run as repeated runs of terminating memory, the outcome of unknown cards was decided by a random number between zero and one, given by Matlabs `rand`, compared to the probabilities of each outcome. The Optimal player faced three different players, one at the time.

Opponents

First, a description of the three opponents the Optimal player faced.

Aggressive child always uses the risky strategy. In this way, the expected number of pairs collected each turn is maximised.

Defensive professor always uses the safe strategy. In this way, the net expected number of pairs in the professors turn is maximised.

Improved professor is using the same strategy as the Defensive professor except when there is only one unknown pair. In that case, the risky strategy is dominating the safe strategy for all outcomes. Then the Improved professor will use the risky strategy.

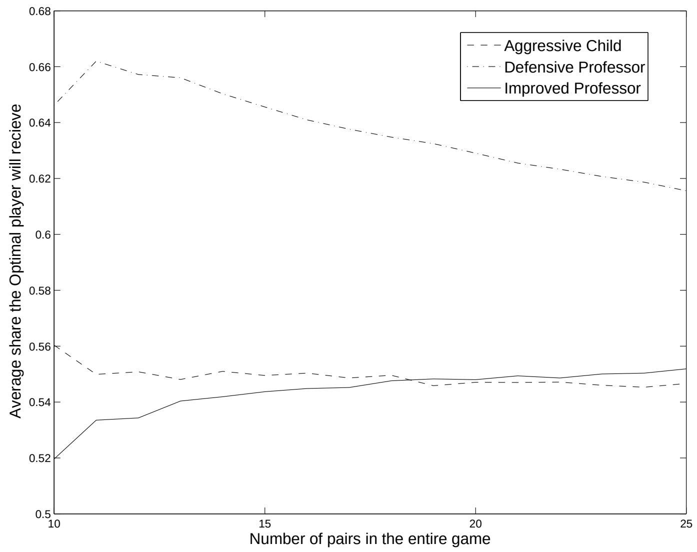

Figure 2.10: The Optimal player does get more than half of the pairs. The Defensive professor loses immense because of his exaggerated cautiousness.

Average outcomes

The optimal player faces each player on boards on 16 different sizes, beginning with from 10 unknown pairs up to 25 unknown pairs. On each size of the board, each player faces the optimal player 100000 times. The average outcome is displayed in figure 2.10. Against all players, Optimal player fares better than the opponent, and collects at least about 55% of the pairs on average. The too cautious Defensive professor loses quite significant amount of pairs, but the difference declines for a larger initial board. This happens since the expected share of the pairs that is delivered from positions $(n, 0)$ declines with growing total number of pairs.

2.5 Conclusion

No general way of determine E_n^j by an explicit expression was found within the scope of this paper. It was found that the optimal strategy must depend on position, there seems to exist some value of parity, positions with an even number of unknown pairs are more favourable than the next odd position with the same number of pairs. Since no way of determine or at least estimate E_n^j was found this could not be resolved. Instead, another game of memory was defined in order to try to find out what can generate such a pattern. The results of this approach are given in chapter 3.
However, clear patterns where detected, which produces two conjectures,

Conjecture 1 Except for smaller boards, $n < 16$, the optimal strategy is to use the risky strategy when j is odd and safe strategy when j is even.
Conjecture 2 Except for smaller j, $j < 18$, $\frac{E_n^j}{n}$ is monotone in n for fixed j, growing if j is even and declining if j is odd,

where the second conjecture probably is needed in order to prove the first. The computed results that resultet in these conjectures are given in section 2.2. The optimal strategy for smaller boards are given by table 2.1 on page 17.

Chapter 3

Combinatorial memory

3.1 What is a combinatorial game?

1. There are just two players.
2. There are several, usually finitely many, positions.
3. There are clearly defined moves that specify the moves that either player can make from a given position to its options.
4. The two players move alternately, in the game as whole.
5. Both players have complete information, that is, all circumstances are known.
6. There are no chance moves.
7. In the normal play convention a player unable to move loses.
8. The rules are such that the play will always come to an end because some player will be unable to move. This is called the ending condition. No game is infinite or drawn by repetition of moves.

Terminating memory is constructed in order to fulfil as many of this rules as possible without changing the children game of memory in any significant manor. However, terminating memory is not a combinatorial game. Rule number 7 can also be stated as the player that makes the last move loses. The game is then called misère instead of normal. Even though this might seem as the exact opposite, i.e. the player that wins the normal game loses the misère game, a game position may be a winning position with both normal and misère rules. One common combinatorial game is the game of nim. Since this will be important to a combinatorial variant of memory, as described in section 3.3, the next section will briefly analyse a variant of nim, called One-heap Nim or Subtraction game.

3.2 One-heap Nim

A throughrough analyse of this game can be found in chapters 2, 4 and 12-17 of Winning ways [1]. In this section, the specific game of nim used in combinatorial memory will be described.
Impartial games, i.e. games where both players are allowed the same moves, given the same game position, can have only two outcome classes, which may be called

$\mathcal{P}$-positions, player in turn is losing, and,

$\mathcal{N}$-positions, player in turn is winning.

No other value is possible for an impartial game. In one heap nim, or subtractions games, the players alternatively removes a number of *beans* from the *heap of beans* of size n. The game ends when there are no beans left to remove. The number of beans a player may remove is given by game specific rules. In this specific game, the allowed moves are to remove 1 or 2 beans. The outcome of the game is given by a function $\mathcal{G}(n)$, given by recursion. In normal game, the player that can reduce the number to 0 wins, since no more beans can be removed. In misère game, that move is the losing move.

3.2.1 Normal game

The position 0 is losing, i.e. the positions from where 0 can be reached are winning. This defines the recursion that determines which positions will be $\mathcal{P}$- and $\mathcal{N}$-positions. If a $\mathcal{P}$-position can be reached, the position is an $\mathcal{N}$-position, otherwise it is a $\mathcal{P}$-position, since it have to reach a $\mathcal{N}$-position and thus is losing.

$$
\begin{array}{rcllllllll}
n & = & 0 & 1 & 2 & 3 & 4 & 5 & 6 & \dots \\
\mathcal{G}(n) & = & \mathcal{P} & \mathcal{N} & \mathcal{N} & \mathcal{P} & \mathcal{N} & \mathcal{N} & \mathcal{P} & \dots
\end{array}
$$

It is a classical result that any position such that $n = 0 \pmod{k+1}$ is a losing position in normal nim if it is allowed to remove any number of beans up to $k \in \mathbf{N}$. The strategy for the player in winning position is of course to force the opponent to start from losing positions. This can not be thwarted, since the steps between losing positions is $k + 1$.

3.2.2 Misère game

The misère game includes the same recursion and thus game mechanics as the normal nim, but the start of the recursion is different. The objective is to not remove the last bean. The only position where any player is forced to do so is when there is only one left. Thus $\mathcal{G}(1) = \mathcal{P}$. The resulting game is

$$\begin{array}{rcllllllll} n & = & 0 & 1 & 2 & 3 & 4 & 5 & 6 & \ldots \\ \mathcal{G}(n) & = & \mathcal{N} & \mathcal{P} & \mathcal{N} & \mathcal{N} & \mathcal{P} & \mathcal{N} & \mathcal{N} & \ldots \end{array}$$

Thus there are positions where the player in turn may win both misère and normal game.

3.2.3 Choice game, used in combinatorial memory

In the game defined in section 3.3, a sequence of subtraction games will occur. Locally they are all misère games, but due to the value of lead[1], a player may win overall by sacrificing the present subtraction game, in order to play in the winning parity. Thus the need to choose to lose occurs. If it is a winning strategy to lose the present misère game, both players will try to do so and vice versa. The choice occurs when the player in turn can win both the misère and the normal nim. Define a new outcome class to the two defined in section 3.2,

$\mathcal{C}$-positions, player in turn may choose whether to win the misère game.

Such positions occur when $n = 2 \pmod 3$, since those positions are winning for both misère and normal nim, where the player is allowed to remove one or two beans. The resulting game is

$$\begin{array}{rcllllllll} n & = & 0 & 1 & 2 & 3 & 4 & 5 & 6 & \ldots \\ \mathcal{G}(n) & = & \mathcal{N} & \mathcal{P} & \mathcal{C} & \mathcal{N} & \mathcal{P} & \mathcal{C} & \mathcal{N} & \ldots \end{array}$$

[1]Defined in section 3.4.3

This game is the one-heap game of nim that will be important in combinatorial memory as defined in section 3.3.

3.3 Making memory a combinatorial game

The game *terminating memory*, described in chapter 2, does not comply with all the 8 rules given in section 3.1. In particular, rules 5-7 is not fulfilled. Even thought perfect memory give the players complete information of the past, the order of unknown cards are, as their name states, unknown. Since the order is not known, it is also not determined, the actual order is an outcome of chance. Finally, in the game of terminating memory, the player that has collected most pair when the game ends is the winner.
However there is an easy change of terminating memory into what henceforth will be called combinatorial memory. The difference is that the order in which the "unknown cards" are to be opened is known to both Left and Right player. Thus both complete information is introduced and chance moves are removed. These changes also turn out to render a game which almost complies with rule 7. The allowed moves will be the same as described in section 2.1.3. The game will turn out to be a sequence of misère nim games. Before that is properly verified, the new game mechanics will be displayed by some examples.

3.3.1 Examples

Henceforth cards in a game are labelled Jx. The numbering of each pair, J, is done from 1 to N,N being the number of pairs in the game, and the labelling $x \in \{a, b\}$ of each card in the pair, where a is used before b. Thus any card Ja is opened before Jb and before any card labelled $J'a$, where $J < J'$. This is only due to labelling and does not change the game, compared to other possible labelling.

An example

Assume the game is a game of 8 pairs where the cards are to be opened in the order displayed by figure 3.1, Left starting the game. As in terminating memory, initially Left has to open two new cards, $1a$ and $2a$, since no known cards can be chosen and every move in memory include opening of two cards. Right has to open card $1b$ and collects pair 1 by then opening card $1a$. Right must continue by opening $3a$. Right player may now choose whether to open $2b$ or $2a$ as his next card, but the outcome will not differ, Right turn will end. Left will collect pair 2 but is then forced to open $4a$. By the same argument as for pair 2, Right will collect pair 3 and will continue and collect pair 4. Right now have to open both card $5a$ and $6a$ since no known cards are to choose from. The need to number the pairs in order now surfaces. Since 8 cards has been opened before $5a$ but there are only 8 possible cards, consisting of pairs 1-4, the previous 8 cards all match to pairs. Thus all pairs and all cards before $5a$ is already removed and can not be chosen. Actually, if any card with label a such that $2(J-1) = M$ where M is the number of previously opened cards, there are no known cards left on the board and the only allowed move forces the player to open 2 cards.

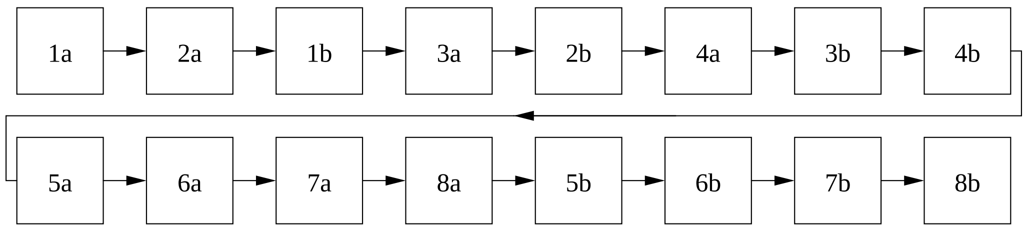

Figure 3.1: A small example of combinatorial memory

Back to the example. Right thus has to open cards $5a$ and $6a$. Left has a choice whether to open $8a$ or not. In this case it is simple, if Left opens $8a$ Right will collect pairs 5-8, otherwise Right have to open $8a$. Thus Left chooses to end his turn, Right gets the lead but has to open $8a$, thus losing the lead. Left gains the lead and will collect pairs 5-8.
Where was the game of Nim?
In this example, the game of Nim occurred five times. Initially, there was a heap of one, Left had to open both $1a$ and $2a$ and Right collected pair 1. $3a$ was a heap of one, $4a$ was a heap of one, $5-6a$ was a heap of one; since no other cards were left; and $7-8a$ was a heap of 2. As stated in section 3.2, in a misère game of one heap nim of size n, where the players may chose to remove one or two, starting player will lose if either player wants it if $n = 1 \pmod 3$ and may choose whether to win or lose if $n = 2 \pmod 3$. Thus Left was able to win the last 4 pairs, since Left started the only Nim-game where the starting player had a choice.
With one major exception it seems like the nim-games are sequences of a-labelled cards. The length of the sequence is equivalent to the size of the heap, n. The exception occurs when there are only non-opened, “unknown”,

cards on the board. This is equivalent to when the first opened card's label is $\{Ja : 2(J-1) = M\}$ where M is the number of opened cards. This exception must be taken care of when the game of combinatorial memory is translated into a sequence of weighted games of nim.

A similar board but a different outcome

Now assume the board given by the sequence shown in figure 3.2. The only difference from the first example is that card $3b$ is moved. This gives two important differences in the game. Most important for game understanding, the nim game consisting of $4a$ is removed, since any player that opens $4a$ as a first move will also be able to open $4b$, collecting a pair. Any player that opens $4a$ as a second card will lose pair 4, but this is equivalent of opening $4b$ as a second card. Thus not all sequences of Ja cards results in nim games. The other difference is that cards $5a$ and $6a$ is not forced to be open as in the first example, card $3a$ is still in play since $3b$ is not opened.

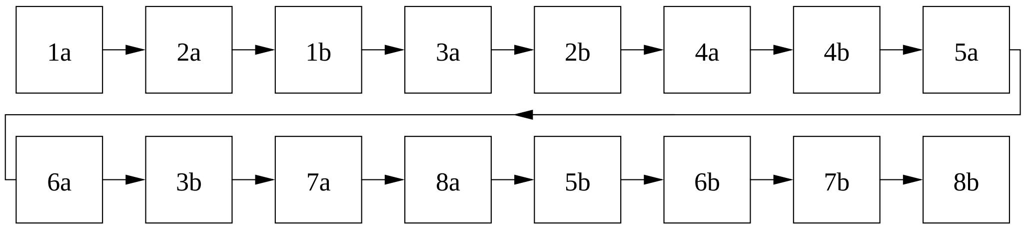

Figure 3.2: Another board of combinatorial memory

Left is forced to open $1a$ and $2a$. Right collects pair 1, and must open $3a$. Left may then collect pair 2, must open $4a$ but may match this with the next card. Thus Left will continue by collecting pair 4 and open card $5a$. Since pair 3 is not collected, Left may and will choose not to open $6a$, since Left then will collect pair 3, after Right opening $6a$, and open card $7a$, forcing Right to open $8a$, leaving pairs 5-8 for Left to collect.
The exception to the general rule of turning a-sequences into n-sized nim heaps in this game regards pair 4. Any pair with the two cards adjacent must be translated into a game with choice of size 1. The choice may however be removed by the same reason as in the first example, there are only new cards left to open with the second card.

Thus the game of combinatorial memory turns out to be equivalent to a sequence of nim games. The translation is rather simple, but some exceptions to the general translation rules arise.

3.4 Deciding winner of a general game

3.4.1 Transforming simple games into one nim game

The example games could be divided into a sequence of alternating misère nim games and collect sequences. Suppose that all games of combinatorial memory can be built up by simple games, consisting of a nim game followed by a collect sequence. With this approach, however, both the objective of winning and *losing* must be analyses, since the winning player also must start the next nim game. Unless otherwise stated in this section, both Safe and Risky strategy is allowed, i.e. there is at least opened, uncollected card left on the board, which is game equivalent to that both one and two may be deducted in the nim game. In any simple game, the game starts with a one

Lead shifting game: Left loses misère nim

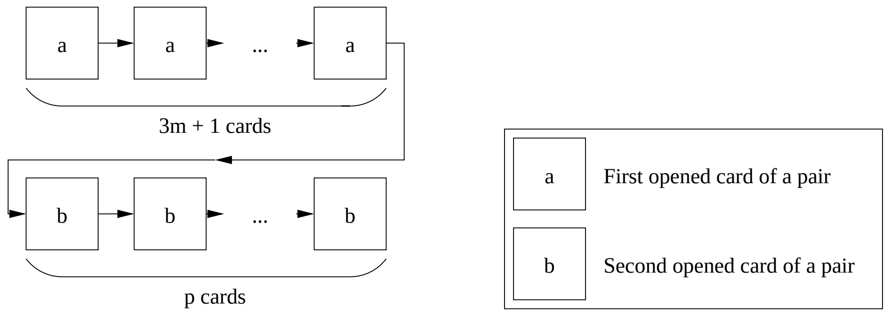

Figure 3.3: A simple game of combinatorial memory where a single misère nim game is of size n s.t. $n = 1 \pmod 3$

heap nim game of size n, where one or two can be deducted.
If $n = 1 \pmod 3$, the starting Left player will lose the misère game and win the normal game if either player wants it. Either Left or Right will make Left lose the nim game, since it must be either good or bad to win the misère nim game. Thus, Left will always open the last card that can not be matched and Right collect the p pairs. Right will have the lead in the next game.

Lead preserving game: Left wins misère nim

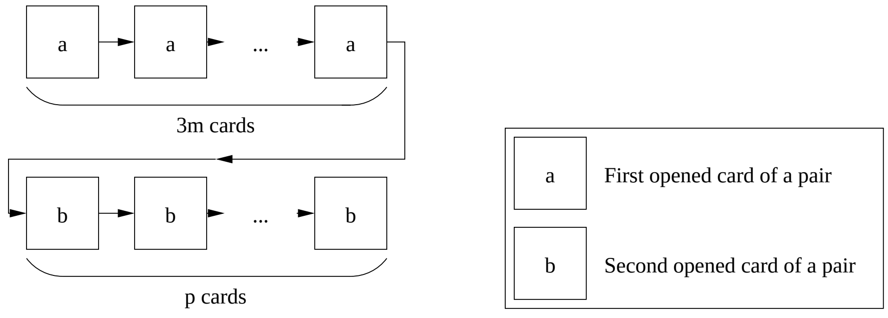

Figure 3.4: If $n = 0 \pmod 3$, the starting Left player will win the misère game and lose the normal game. Thus, Right will open the last card that can not be matched and Left collect the pairs.

Choice of lead game: Left optionally wins misère nim

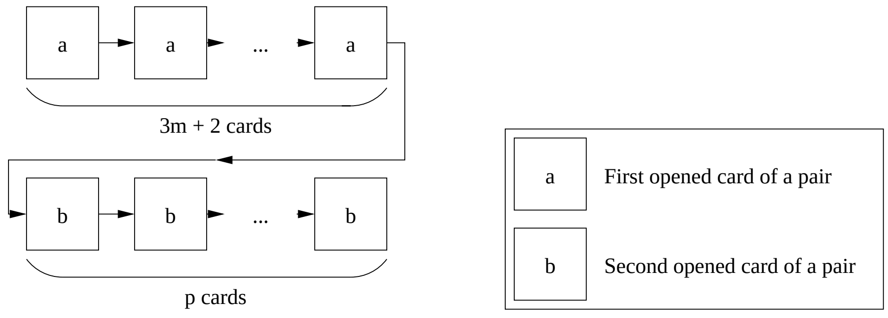

Figure 3.5: If $n = 2 \pmod 3$, the starting Left player will be able to win both the misère game and the normal game and can thus choose whether to collect the pairs and start the next nim game or not.

Entire game, example of no choice case

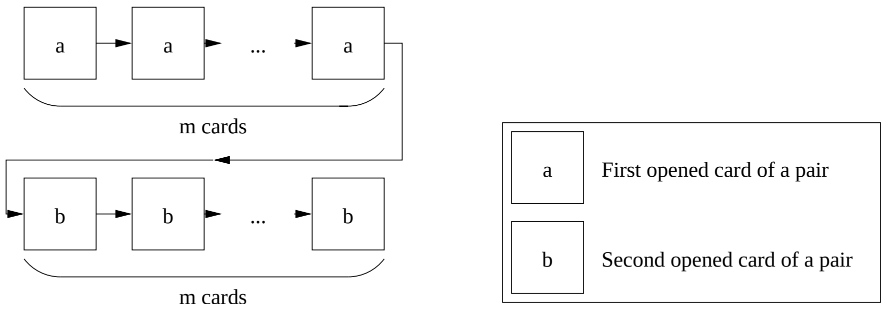

Figure 3.6: If there are no known cards on the board, the Safe strategy, i.e. open only one new card, is not allowed. This happens if and only if the first opened card is labelled Ja where $\{J : 2(J-1) = M\}$, M being the number of previously opened cards. This is equivalent with $\#a$-cards $= \#b$-cards. Then the two cards which Left player must open may be translated as a separate nim game, with $n = 1$, with no pairs to collect, i.e. a zero worth type of simple game of lead shifting nim. This is true since Left must open the two initial cards and end his turn, unless the case of an adjacent pair is valid.

Adjacent pair case

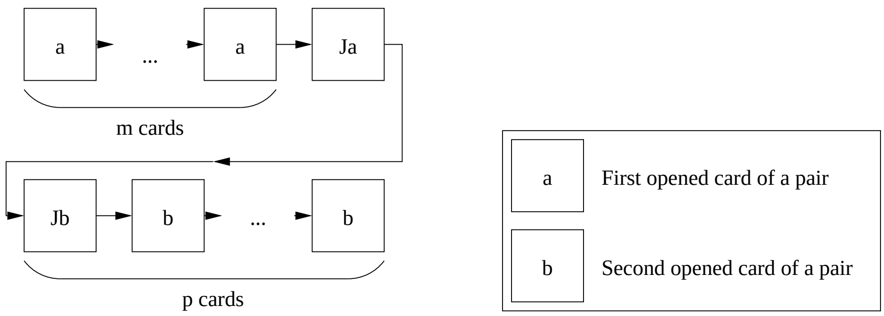

Figure 3.7: If any card Ja is followed by card Jb, the Ja is not counted as a part of the nim game, since the player that opens Ja with his first card, may collect pair J. Translate any such pair into a one pair worth choice of lead game. However, this case can be combined with the no choice case. The two cards are then translated into a lead preserving game of size one.

3.4.2 Transforming any game into a sequence of nim games

If any board can be translated to several simple games, the general game of combinatorial memory could be solved as a sequence of misère nim games. This might be good since nim games are easier and more known than combinatorial memory.

Theorem 1. *Any game of combinatorial memory can be decided as a sequence of nim games*

Proof. If dividing the board into simple games of the three types is possible for any board of combinatorial memory, assuming that the two exceptions are translated into stated variants of simple games, the theorem is true. Assume there is a sequence of some board that can not be described. No cards in this sequence labelled a can be before any card labelled b, in that case there would be a nim game followed by a collect sequence, describe by one of the simple games. But in any legal sequence of combinatorial memory, there must be a card labelled a first. Either it is the first subsequence in the game, and the first card in the game is labelled $1a$, or it must be separated from a previous collect sequence. Likewise, the last card in the sequence must be labelled b. Either it is the last card in the whole game, that must be the second half of a pair, or a b labelled card must separate this sequence from the initial a labelled card of the next sequence. Thus any part of a board that can not be translated into a series of weighted nim games is not consistent with combinatorial memory. □

3.4.3 Adding of memory nim games

Notation

Lead: The player that starts the nim game is said to have the lead because that player collected the pairs of the previous nim game.

Lead shifting game: does shift lead, Right player (of the misère nim) will collect the following p pairs. This will be noted $\downarrow p$.

Lead preserving game: does not shift lead, Left player (of the misère nim) will collect the following p pairs. This will be noted $\uparrow p$.

Choice of lead game: does shift priority iff Left player (of the misère nim) gains from it, Left decides which player will collect the following p pairs. This will be noted $\updownarrow p$.

Also notice that since a net game result is computed as the number of pairs Left will collect deducted the number of pairs Right will collect, the number of pairs p may be negative, i.e. starting player collects fewer pairs than the second player.

Lemma 1. *Adding a lead preserving nim game yields*

$$\begin{aligned} \uparrow p_0 \uparrow p_1 &= \uparrow (p_0 + p_1) \\ \downarrow p_0 \uparrow p_1 &= \downarrow (p_0 + p_1) \\ \updownarrow p_0 \uparrow p_1 &= \updownarrow (p_0 + p_1) \end{aligned}$$

Proof. The player that gets the p_0 pairs will automatically also get the p_1 pairs, since either Left or Right will enforce it, depending on the sign of p_1 and the value of the lead. □

Lemma 2. *Adding a lead shifting nim game yields*

$$\begin{aligned} \uparrow p_0 \downarrow p_1 &= \uparrow (p_0 - p_1) \\ \downarrow p_0 \downarrow p_1 &= \downarrow (p_0 - p_1) \\ \updownarrow p_0 \downarrow p_1 &= \updownarrow (p_0 - p_1) \end{aligned}$$

Proof. The player that gets the p_0 pairs will automatically also lose the p_1 pairs, since either Left or Right will enforce it, depending on the sign of p_1 and the value of the lead. □

Since adding of a lead shifting nim game deducts the net number of pairs the player in the lead will get, negative p_i may occur.

Theorem 2. *Any sequence of memory nim games with one initial choice of lead game and no other games with choice of lead, is game equivalent of a single nim game* $\updownarrow q$.

Proof. This is almost a corollary of Lemma 1 and Lemma 2, since any addition of two non-optional games returns a non-optional nim game. Thus, any sequence of nim games without choice can be summed together to a single nim game without choice. The initial choice of lead game can then be added with the acquired game. □

Lemma 3. *Adding a game with choice of lead from the end (where lead is unimportant) yields*

$$\begin{aligned} \uparrow p_0 \updownarrow p_1 &= p_0 + |p_1| \\ \downarrow p_0 \updownarrow p_1 &= -(p_0 + |p_1|) \\ \updownarrow p_0 \updownarrow p_1 &= |p_0 + |p_1|| \end{aligned}$$

Proof. The player that gets the p_0 pairs may choose whether to give or take the p_1 pairs, depending on the sign of p_1. However, since it includes a choice, which player will get the lead is not clear. Thus it is necessary to add the games together from the end of the game, where the lead is without value, there is no more pairs to collect. □

Theorem 3. *Any sequence of choice of lead nim games can be solved by recursion.*

Proof. Since any two $\updownarrow$ p-games can be added when the lead does not matter, this follows from Lemma 3. In order to use the same notation, some of the addition of Lemma 3 has to be reverted, the choice of sign in the outermost absolute sign can be noted with a $\updownarrow$ without losing consistancy. The ownership of the lead is unknown but also irrelevant.
Recursive step, no games will be added from the right.

$$\updownarrow p_{m-2} \updownarrow p_{m-1} \updownarrow p_m = \updownarrow p_{m-2} \updownarrow (p_{m-1} + |p_m|)$$

□

Corollary. *Since any game of combinatorial memory is game equivalent of a sequence of choice of lead nim games, it can be resolved by recursion as stated in Theorem 3.*

This follows when combining all three theorems. For any board of combinatorial memory, the winner can be decided by translating the game in to a sequence. The sequence can be simplified by the adding rules stated in Lemmas 1-3, but since adding of optional shifting game includes an absolute value function and does not include information of the ownership of the lead, this will be nested if there are more than one optional shifting game.

3.5 Translating the examples

Return to the example given in section 3.3.1. It can be translated into the following sequence,

$$\downarrow 0 \uparrow 1 \downarrow 1 \downarrow 2 \downarrow 0 \updownarrow 4$$

Before any simplification is done, p sums to 8, the number of pairs. By the addition rules, this simplifies to

$$\begin{array}{lllll}
\downarrow 1 & \downarrow 1 & \downarrow 2 & \downarrow & \updownarrow 4 \\
\downarrow 1 & \downarrow 1 & \downarrow 2 & \downarrow 4 & \\
\downarrow 1 & \downarrow 1 & \downarrow -2 & & \\
\downarrow 1 & \downarrow 3 & & & \\
\downarrow -2 & & & & \\
2 & & & &
\end{array}$$

Which means Left wins by two pairs, since Left starts. The result is then $5 - 3$, which confirms the result given the first time this example was used. In the same way the example given in section 3.3.1 can be translated to

$$\downarrow 0 \uparrow 1 \downarrow 1 \updownarrow 1 \updownarrow 1 \updownarrow 4$$

which in turn can be simplified to

$$\begin{array}{lllll}
\downarrow 1 & \downarrow 1 & \updownarrow 1 & \updownarrow 1 & \updownarrow 4 \\
\downarrow 1 & \downarrow 1 & \updownarrow 1 & \updownarrow 5 & \\
\downarrow 1 & \downarrow 1 & \updownarrow 6 & & \\
\downarrow 0 & \updownarrow 6 & & & \\
6 & & & &
\end{array}$$

In this case Left wins by 6 or $7 - 1$, one can recollect that Right managed to collect only pair 1. This, however, could be somewhat easier solved if the theorems of section 3.3.1 are used.

$$\downarrow 0 \uparrow 1 \downarrow 1 \updownarrow 1 \updownarrow 1 \updownarrow 4$$

is immediately transformed into

$$0 \updownarrow 1 \updownarrow 1 \updownarrow 4$$

from where the recursion gives the same as above

$$|1 + |1 + |4||| = 6$$

If the games are larger, this is more useful. Assume the board to be game equivalent to

$$\updownarrow 5 \downarrow 3 \uparrow 1 \updownarrow 3 \uparrow 4 \updownarrow 1 \uparrow 3 \uparrow 4$$

By using Lemmas 1 and 2, this is equivalent to

$$\updownarrow (5 - (3 + 1)) \updownarrow (3 + 4) \updownarrow (1 + 3 + 4)$$

which, when recursively computed gives

$$|1 + |7 + |8||| = 16$$

As can be seen, the recursion can be exchanged with a nested absolute function, perhaps simpler than pure recursion when using computer aided computations.

3.6 Further studies

In order to make these results useful to determine strategy of terminating memory, statistics of possible remaining boards must be determined in an easy way. Since a-labelled cards can be placed anywhere on the board, but b-labelled cards must be later than the matching a-card, the size p of the collect sequences are more likely to be larger in the end game. It is also most likely that the most probable nim game size is 1, i.e. the players can expect to get every other collect sequence. However if the number of expected collect sequences is even, the player would try to give the next sequence away in order to be able to claim the last, most likely biggest, collect sequence. If the number is odd, the player will try to get the next collect sequence in order to get one more collect sequence, including the last.
To make this hypothesis consistent with the results of chapter 2, one has to confirm that when the number of unrevealed pairs j is odd, it is more likely to be an even number of collect sequences left. Corresponding, it must be more likely to be an odd number of collect sequences left when j is even.
The risky strategy would give away the collect sequence if the size of the current nim game is 2, which is less likely than a size of 1 but more likely than a size of 3. Since strategy does not matter when the size is 1, a nim game of 2 must be expected when choosing strategy. Corresponding, the safe strategy increases the possibility for the player in turn to get the next collect sequence.
This is only a hypothesis that might be confirmed by further studies.

Appendix A

Expected value function E_n^j

In the appendix, the computed function E_n^j is given for all n, $j < 23$. They are stated as 23 functions in n, one for each fixed, given j.

$$
\begin{aligned}
E_n^0 &= n \\
E_n^1 &= \frac{n+2}{3} \\
E_n^2 &= \frac{11n-12}{15} \\
E_n^3 &= \frac{13n^2+119-180}{35(n+3)} \\
E_n^4 &= \frac{211n^2+72n+1232}{315(n+4)}
\end{aligned}
$$

From $j = 5$, the general pattern of strategy choice is broken. Each element can be computed seperatly, but to solve the general recursion the strategy has to be the same for all n.
However, the pattern is reestablished for $n \geq 16$. Thus, henceforth the expected value functions are valid only for $n \geq 15$, where the value for $n = 15$ is used as initial value. For E_n^5 and E_n^6, the expected value function can be generally computed from $n = 9$ instead, since the pattern has not get broken for these j.

Unless otherwise stated, the expected value functions are valid only for $n \geq 15$.

$$
\begin{aligned}
E_n^5 &= \frac{1}{3454(n+5)(n^2-16)(n^2-9)(n^2-4)(n^2-1)} \times \\
&1355n^{11} + 22517n^{10} - 208780n^9 + 274890n^8 + 1612215n^7 - 7158459n^6 + 6211810n^5 + 28105660n^4 - 47631320n^3 \\
&-21244608n^2 + 40014720n + 1207722700800 \quad n \geq 9 \\
E_n^6 &= \frac{1}{45045((n+6)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n)} \times \\
&28785n^{13} - 40066n^{12} + 640809n^{11} - 12622610n^{10} - 18740865n^9 + 403799682n^8 - 168861693n^7 - 3741258950n^6 + 3842132580n^5 + 11313085784n^4 - \\
&-14329401216n^3 - 94210333626240n^2 + 847832010163200n - 999698373273600 \quad n \geq 9 \\
E_n^7 &= \frac{1}{45045(n+7)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \times \\
&18179n^{15} + 456901n^{14} - 9671221n^{13} + 73737209n^{12} - 322893571n^{11} - 403780377n^{10} + 13170694537n^9 - 39346766173n^8 - 85591846340n^7 + \\
&+500442798856n^6 - 159474385616n^5 + 328040918687664n^4 - 8570817229963968n^3 + 72611046391461120n^2 - 215092266770304000n + 1360915309907112960 0 \\
E_n^8 &= \frac{1}{11486475(n+8)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \times \\
&7123515n^{17} - 29253600n^{16} + 1121029436n^{15} - 29510640000n^{14} + 122122097090n^{13} + 1146119083200n^{12} - 8658956028268n^{11} - \\
&-2157444432000n^{10} + 150474833240875n^9 - 310556354565600n^8 - 725126934312872n^7 - 220762956566880000n^6 + 8204274258706790320n^5 - \\
&-1136444651656721280 00n^4 + 728041184240995816704n^3 - 2949366570456314880000 0n^2 + 3583336363898769580032 00n + 1574744966567395983360 00 \\
E_n^9 &= \frac{1}{72747675(n+9)(n^2-64)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \times \\
&30006585n^{19} + 1036841875n^{18} - 39238112922n^{17} + 668210240988n^{16} - 8441611673826n^{15} + 53357036901330n^{14} + 100382189973576n^{13} - \\
&-3236618760775244n^{12} + 12525972644244033n^{11} + 28307835115360515n^{10} - 304205343026273286n^9 + 3630213923571979224n^8 - \\
&-199280597740024684128n^7 + 5124649653990178853680n^6 - 68552499294181790489568n^5 + 129503402529960908398003 2n^4 - \\
&-2868415531658423336785766 4n^3 + 2570513389366492045817856 00n^2 - 3309407156257467023904768 00n - 2201148062383258856964096 000 \\
E_n^{10} &= \frac{1}{43648605(n+10)(n^2-81)(n^2-64)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \\
&26501985n^{21} - 203623026n^{20} + 13034913771n^{19} - 528451426582n^{18} + 6391487629098n^{17} - 13173868153668n^{16} - 332091428099706n^{15} \\
&+3083965598740596n^{14} - 6073575327255579n^{13} - 76008187869433690n^{12} + 545890469445740367n^{11} - 4091982911069169246n^{10} + 27825459095445911222 4n^9 \\
&-9037373963522997668112n^8 + 156919129912520606967600n^7 - 3184376290476848339462368n^6 + 8602943299654806013761907 2n^5 \\
&-1313125210862475835623381504n^4 + 7647487749569929420200019968n^3 + 9658785159141709071159705600n^2 - 2371151985874325566864293888 00n \\
&+6104195254560862619798077440 00
\end{aligned}
$$

$$
\begin{aligned}
E_n^{11} &= \frac{1}{35137127025(n+11)(n^2-100)(n^2-81)(n^2-64)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \times \\
& 14730598575n^{23} + 659637222405n^{22} - 39161256924075n^{21} + 1160590782054011n^{20} - 26812896242751850n^{19} + 378492647435609630n^{18} \\
& -2309093970255831990n^{17} - 83390447333797779034n^{16} + 227250401966888216235n^{15} - 12236426230520582773295n^{14} - 16320401591880983454555n^{13} \\
& +53026478042819655976511n^{12} - 790471125202981847412480n^{11} + 29500872418858199679515580n^{10} - 805274023543443003827740320n^{9} \\
& +17249987481159244162110603536n^{8} - 427412152811575702015317022720n^{7} + 9595469883147853800892136943680n^{6} - 132376912961774532965825060540160n^{5} \\
& +863829057066740976487619677694976n^{4} + 72673963511954793009295861002240n^{3} - 34974009638426961315345065084928000n^{2} \\
& +188307942156761715586869163794432000n - 337665229522362661804740397301760000 \\
E_n^{12} &= \frac{1}{35137127025(n+12)(n^2-121)(n^2-100)(n^2-81)(n^2-64)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \times \\
& 20995752393n^{25} - 252722296680n^{24} + 23977745723430n^{23} - 1415269186258032n^{22} + 31981440446585047n^{21} - 354278944946434200n^{20} \\
& +1216153342920445160n^{19} + 24386270068074750528n^{18} - 398997643370792126793n^{17} + 1688168795113163036520n^{16} + 13116424253053533069790n^{15} \\
& -167521280587897574636592n^{14} + 1434411445793289327582617n^{13} - 61612445696276780946672360n^{12} + 2039607468392158476472410420n^{11} \\
& -49945386029922312877453901472n^{10} + 1355009824352680665761046578992n^{9} - 35953676244797355350995701344640n^{8} + 654033498243091716930476625560000n^{7} \\
& -6624496970848860540456096080543232n^{6} + 20269877847795994690568373630879744n^{5} + 304693904420829370779162604093071360n^{4} \\
& -4080408171629814956368568781681868800n^{3} + 22251618529360531526985672892140748800n^{2} - 62534485757543742842514559385075712000n \\
& +79314475585750982172580876139888640000 \\
E_n^{13} &= \frac{1}{175685635125(n+13)(n+12)(n^2-121)(n^2-100)(n^2-81)(n^2-64)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \times \\
& 74594975455n^{26} + 5053069157325n^{25} - 296053780804308n^{24} + 12674401215167894n^{23} - 433416896087588147n^{22} + 8686494141819477731n^{21} \\
& -89697058796791682698n^{20} + 160372144362332705264n^{19} + 7879896013707917968593n^{18} - 96876627498917524651789n^{17} + 271988847408244604090992n^{16} \\
& +3758782918344388721130494n^{15} - 46678104062391243723489397n^{14} + 883707183681529905155409581n^{13} - 33096135776509781581588962258n^{12} \\
& +962344788435534516276407391644n^{11} - 26420597049992306198365127933672n^{10} + 751668321599361406312011736056656n^{9} \\
& -17051852317574628994609864445340128n^{8} + 247308181990428290806687900698576704n^{7} - 1801095998265271216696614895442696832n^{6} \\
& -2397941914258247568460042153305429504n^{5} + 177944554894330790321019910459717478400n^{4} - 1691739675329446048724636848669043712000n^{3} \\
& +8204402817361110078076149393459830784000n^{2} - 22056233324658138843217068069160058880000n + 27959306145949011831635526481693900800000
\end{aligned}
$$

$$E_n^{14} = \frac{1}{32752821976875(n+14)(n+13)(n+12)(n+11)(n^2-100)(n^2-81)(n^2-64)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \times$$
$$19325726394975n^{26} + 365202175373550n^{25} + 48174021307779130n^{24} - 1919776742572753620n^{23} + 36776600478626000041n^{22}$$
$$-456419771843533498374n^{21} - 3591829200593843316 80n^{20} + 926181456280116041227 20n^{19} - 825648302319218169518639n^{18} -$$
$$2913448739290151872861854n^{17} + 766449012478992618713730 10n^{16} - 243334611861002425919747940n^{15} + 904990473732399664098165271n^{14}$$
$$-1970293806781247036149926155 94n^{13} + 7736225430703531321911086960500n^{12} - 20899655175612668450927793822 1800n^{11}$$
$$+649362090211930288320676546354 0336n^{10} - 2080856439979006247975570797556975 04n^{9} + 452864978881073279614453414316 6895040n^{8}$$
$$-52115027459553642727898214164082111360n^{7} + 975536493828114342279103600989506780 16n^{6} + 5450931036467922658675044612333674059776n^{5}$$
$$-78241567629949428020910496848383026176000n^{4} + 536935757738168511376812199444898598912000n^{3} - 2205582280577783606416411044924218818560000n^{2}$$
$$+5812654834598461694331791156630476554240000n - 8335208216480595004315984091172175872000000$$

$$E_n^{15} = \frac{1}{1421472473796375(n+15)(n+14)(n+13)(n+12)(n+11)(n^2-100)(n^2-81)(n^2-64)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n} \times$$
$$609791965207425n^{27} + 71576234194899105n^{26} - 1803931020327014040n^{25} + 15865394161755254847 0n^{24} - 590501492534078693994 1n^{23}$$
$$+108261160100620722844111n^{22} - 937153463358280963906298n^{21} - 5611431318597284317531240n^{20} + 22975067632075461585149 7519n^{19}$$
$$-164746988813156555217443436 9n^{18} - 874258055276143512478779734 8n^{17} + 178337063668877787882778051150n^{16} - 714172914419029956477868083011n^{15}$$
$$+1660045548261540702427134514 6241n^{14} - 1016526637457293494817146523927818n^{13} + 339242636560965530366490939498815 40n^{12}$$
$$-102635952763293982915312966265070 5896n^{11} + 33939740335769714586517760969276588656n^{10} - 927967837957347600068916006649994920928n^{9}$$
$$+1609858443028175188728636971642315511 0080n^{8} - 131903476638937409286688149048582982356096n^{7} - 532685718715874802469998579058661430303744n^{6}$$
$$+2539096827502917626454272114949743950158 6432n^{5} - 28857231449023757502051831678582713550336 0000n^{4}$$
$$+181613748010817139518501208235011206274048 0000n^{3} - 727733245064272335799489093957989177544704 0000n^{2} +$$
$$193810156120423742089796521722695455476940800 00n - 2794798568550432591774496510233384753561600 0000$$

$$E_n^{16} = \frac{4264417421389125(n+16)(n+15)(n+14)(n+13)(n+12)(n+11)(n+10)(n^2-81)(n^2-64)(n^2-49)(n^2-36)(n^2-25)(n^2-16)(n^2-9)(n^2-4)(n^2-1)n}{\times}$$
$$2490477158967525n^{27} + 129927181965584535n^{26} + 1376421831072576225 0n^{25} - 233168468091262922454n^{24} + 8931997190114762206899n^{23}$$
$$-115579102913906447354439n^{22} - 1343854609376705268676900n^{21} + 35492497181097944956068408n^{20} - 128342353087523081610943301n^{19}$$
$$-3144468927752979984522647799n^{18} + 34393730315170954713744766410n^{17} + 716949497630472196513496308 98n^{16} - 1415740242834841505603826410611n^{15}$$
$$-8444798202665995613873416266 8169n^{14} + 38295200039514624438117127579439 20n^{13} - 11209271023599506909474889991270405 2n^{12}$$
$$+371730429495028705640791813470830342 4n^{11} - 139673363995535815609872856595797311984n^{10} + 373770865983250259453722228898063980928 0n^{9}$$
$$-5308888524955540961536631305277380642195 2n^{8} + 12257620348210681401431841619485884986854 4n^{7} + 835227085725211989085643449940567754784665 6n^{6}$$
$$-1468926778268840617314966372073716860564889 60n^{5} + 124547325839871654505106920110976244917252915 2n^{4}$$
$$-661355754141087186310549668303330666815029248 0n^{3} + 263202185581838638089364176014780618099392512 00n^{2}$$
$$-820432846495596507696841807171789525258076160 00n + 136827505874914083603846736312762134712811520000$$

Since the coefficients are growing to ridiculous size, no further computations are appended. No apparent change in the computed results appears for larger j, the concluded conjectures are true and the expressions are big and cumbersome.

Bibliography

[1] Berlekamp, Elwyn R., Conway, John H. and Guy, Richard K. [2001], *Winning ways: for your mathematical plays*, 2nd edition (A K Peters Ltd., 2001-4): ISBN 1-56881-130-6, ISBN 1-56881-142-X, ISBN 1-56881-143-8, ISBN 1-56881-144-6.

[2] Stanley, Richard P. [1997], *Enumerative combinatorics* 2nd printing (Cambridge University Press, 1997): ISBN 0-521-66351-2.