Multiplayer Bulgarian Solitaire

Emil Öhman
emil.t.ohman@gmail.com

under the guidance of
Henrik Eriksson
School of Computer Science and Communication
Royal Institute of Technology

Research Academy for Young Scientists
July 11, 2012

Abstract

In Bulgarian solitaire a deck of cards is divided into piles. One card is then taken from each pile and put together to form a new pile. This process of creating a new pile by removing cards is repeated until some position has appeared twice i.e. the solitaire has entered a cycle.

This paper studies a generalization of Bulgarian solitaire to several players. Properties of this generalization are studied such as the positions belonging to a cycle, possible cycle lengths, number of cycles and its Garden of Eden positions. The paper also notes properties of an inverse version of Bulgarian solitaire. The paper is written with the ambition to be comprehensible for a high school student.

Contents

1 Introduction

The story of Bulgarian solitaire began on a train journey in Russia around 1980, as a mystic stranger showed the mathematician Konstantin Oskolkov a card trick. The man gave Oskolkov a pile of 15 cards and asked him to divide it into smaller piles in any way he liked. Oskolkov decided to divide the 15 cards into five piles of sizes (3, 1, 4, 1, 6). When he was done, the man asked him to take one card from each pile, put these in a new pile and to then repeat this process. The first time, Oskolkov got the piles (3–1, 1–1, 4–1, 1–1, 6–1, 5) $= (2, 3, 5, 5)$ which were followed by (1, 2, 4, 4, 4). By repeating this process he eventually ended up with the piles (1, 2, 3, 4, 5), which were followed by themselves. "That's what always happens", said the man and smiled. Amazed, Oskolkov tried to start with all different kinds of piles, including 15 piles of 1 card each. Still, he ended up with the piles 1, 2, 3, 4 and 5, over and over again.

The story above is true, although its details are fabricated [1]. It turns out that all numbers that can possibly be divided into piles of size 1, 2, 3, $\ldots n$ eventually ends up in exactly those piles [2]. For $n = 10$, this means that you can divide the 55 cards into *any* of its 451 276 possible pile stacks and always end up with piles of size 1 to 10 [3]. How can this be? How long time could it take? What happens for other numbers? Finding solutions to these problems is far from trivial.

It was only a matter of time before the problems would catch the interest of numerous mathematicians. When Martin Gardner wrote an intriguing introduction to the game in the Scientific American, it spread all over the world. However, a few persons gave solutions before Gardner wrote about the game; the first seems to have been Andrei Toom or Borislav Bojanov in 1981 [1]. Another person published a solution in 1981, Henrik Eriksson of the Royal Institute of Technology in Sweden [4]. It was Eriksson who coined the name Bulgarian solitaire, but unfortunately the game turned out to be neither Bulgarian nor a real solitaire.

At a glance, Bulgarian solitaire might seem too simplistic to be applicable to the real world. However, the Brazilian mathematician Serguei Popov has shown that even if cards are excluded at random, the outcome is similar [5]. This means that Bulgarian solitaire could roughly describe the distribution of anything that is steadily accumulated from several smaller entities to form a new entity. An example could be the distribution of resources over departments, where an organization repeatedly needs money for a new department. The organization could provide for these new departments by cutting down on the resources of other departments more or less equally.

A perhaps more frequently encountered system is where a large accumulated entity is distributed over several other smaller entities. This can be described by a dual version of Bulgarian solitaire, where the largest pile is distributed over the others. Such a game has as far as we know never been studied. The dual version might seem like the inverse of the original, which is indeed true regarding certain aspects, however dual Bulgarian solitaire is surprisingly similar to its original. Some of its properties will be noted in this paper.

The main purpose of this paper is to study a generalization of Bulgarian solitaire and its properties. The generalization is best described as a multiplayer version of the original, where several players sit around a table, playing their own instance of Bulgarian solitaire. The players take one card from each of their piles to form a new pile and pass it on to the player on their right. We study how this generalization converges, the maximum number of moves before convergence, the number of cycles, their lengths and the properties of a Garden of Eden position.

The proofs are written with the ambition to maintain a pedagogic approach, allowing another high school student with basic knowledge in discrete mathematics to follow the given proofs.

2 Standard Bulgarian solitaire

2.1 Terminology and notation

The *partitions* of a positive integer are the ways it can be expressed as a sum of positive integers, including the number itself, without considering their order. The partitions of 4 are $(1+1+1+1)$, $(2+1+1)$, $(2+2)$, $(3+1)$ and (4). Piles in Bulgarian solitaire form a partition of the total number of cards.

λ, the lambda sign, is used to denote a partition. In this paper λ is regarded a tuple of terms in decreasing order so that the first term is the largest. The number of terms in a partition is denoted $|\lambda|$. This means that

$\lambda = (\lambda_1,\ \lambda_2,\ \lambda_3\ \ldots\ \lambda_k : \lambda_1 \geq \lambda_2 \geq \lambda_3\ \ldots \geq \lambda_k),\ k = |\lambda|$.

A *Young diagram* is a way to visualize partitions and the distribution of cards in Bulgarian solitaire. 7 cards distributed into the piles 4, 2 and 1 can be described as a partition of 7 since $4 + 2 + 1 = 7$, shown in figure 1.

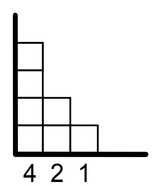

Figure 1: Example of a *Young diagram* displaying a partition of 7.

A *triangular number*, denoted T_n, is defined as $1+2+3+\ldots n$. They are called triangular because they can form a triangle, such as the 10 pins in bowling. The n first triangular numbers are $1, 3, 6, 10, 15, \ldots T_{n-1}, T_n$. For an example $n = 5 \Leftrightarrow T_n = 15$.

In Bulgarian solitaire an operation is repeatedly performed on λ. The operation consists of decreasing all terms in λ with 1, adding a new term whose size is equal to $|\lambda|$ and then removing any zero term. The new term is denoted λ_{new}.

The partition after i operations is denoted λ^i. This means that λ^0 is the initial partition and λ^1 the partition after 1 operation.

2.2 An example

Take 7 cards and arrange them into piles of size 2, 2, 2 and 1. Since $2+2+2+1=7$, the piles can be considered a partition of 7. $\lambda^0 = (2,\ 2,\ 2,\ 1)$.

Remove one card from each pile, and put them in a new pile. You now have one pile of 4 cards and three piles of 1 card. $\lambda^1 = (4,\ 2-1,\ 2-1,\ 2-1,\ 1-1) = (4,\ 1,\ 1,\ 1)$ and $\lambda^1_{new} = 4$. Repeating this process will give the following partitions:

$$2221 \rightarrow 4111 \rightarrow 43 \rightarrow \underline{322} \rightarrow 3211 \rightarrow 421 \rightarrow 331 \rightarrow \underline{322}.$$

As soon as we reach 322 the second time, we know that we are in a cycle, since we have seen this partition before. In this case $\lambda^3 = \lambda^7 \Rightarrow$ we are in a cycle of length 4. Note that 7 can not end up in piles of size $1,\ 2,\ 3 \ldots n$ because it is not a triangular number.

2.3 The cradle model

A brilliant visualization of Bulgarian solitaire was suggested by Anders Björner[1]. We will refer to this visualization as the cradle model. If you turn a Young diagram counter-clockwise by 45 degrees, the Bulgarian operator can be regarded as moving all blocks one step to the right as shown in figure 2. When you remove the bottom block from all piles, each remaining block will move one step to the southeast. When the removed blocks are added as a new pile,

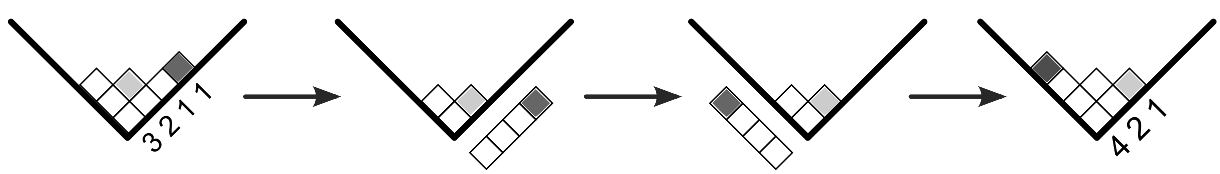

Figure 2: Anders Björner's visualization of a move in Bulgarian solitaire.

[1]Personal communication with Henrik Eriksson 2012.

each block moves one step to the northeast. Since north and south cancel out, one round of Bulgarian solitaire moves each block one step to the east (right). If a block cannot move further to the right, it goes to the far left. Thus, the blocks seem to cycle on the same level.

Consider the move shown in figure 3. If we let all blocks fall towards the left axis, the excessive blocks of larger piles will move towards it. Hence, the sole result of letting the blocks fall down is to sort the piles from largest to smallest. Since order does not matter in Bulgarian solitaire, we can safely apply gravity to our model.

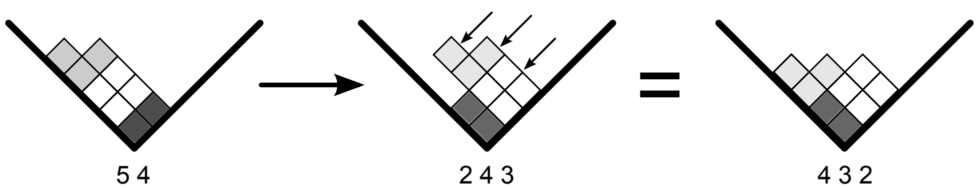

Figure 3: Applying a gravitational pull towards the left axis sorts the piles.

2.4 Convergence

Bulgarian solitaire is completely deterministic, a specific partition always leads to another specific partition. For a given number of cards there is only a finite number of partitions. This means that the game will eventually return to a previous partition. Hence, Bulgarian solitaire always ends up in a cycle (the length of the cycle can be 1 as in $3, 2, 1 \rightarrow 3, 2, 1$). When the game has entered a cycle, we say that the game has *converged*.

We will now prove the conjecture stated in the introduction that all numbers that can possibly be divided into piles of size 1, 2, $3, \ldots$ n (triangular numbers) end up in exactly those piles. This has already been proven in e.g. [2] [4] but the proof has been included for the benefit of the reader.

Theorem 1. *When the total number of cards is triangular, Bulgarian solitaire will converge into piles of size* $1, 2, 3, \ldots, n$, $n \in \mathbb{N}$.

Proof. When a block falls due to gravity, a *hole* is filled. Before being filled, such a hole cycles from left to right as the game goes on, exactly like a block. However, there is one vital difference between a block and a hole; any block situated directly above a hole will take its place. Consider a block moving one level above a hole as shown in figure 4. We can quickly

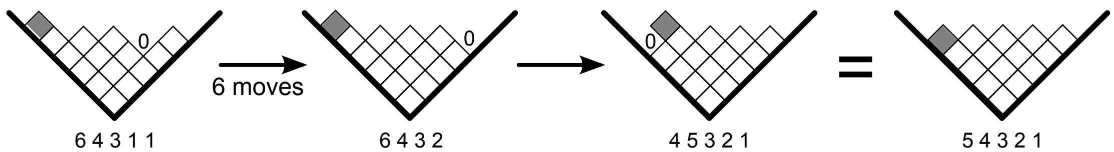

Figure 4: Since the hole (zero) has a shorter cycle it will eventually catch up with the extra block, which will then fall.

see that it will require 6 operations to move the block back to its original position. Since the hole moves one level below the block, it will require 5 operations before it reaches the same position. This means that every 6 moves, the hole will have caught up with the extra block by 1 position; the distance between hole and block is steadily decreasing. When the hole catches up with the block, the block will fall down and take the place of the hole.

More generally, any hole on a level below a block will eventually be filled. This means that after a finite number of moves, all blocks will be as far down as possible; which is attained when the blocks are in a triangular partition. □

Theorem 2. *In the general case with* $T_n + r$ *blocks (*T_n *being as large as possible), the solitaire will converge with the triangular partition as bottom and* r *surplus blocks cycling above. This has been proven in [2] with a more advanced mathematical apparatus.*

Proof. It follows from theorem 1 that eventually there will be no hole below a block, and that no block can fall down if there is no hole below it. When there are no holes left to be

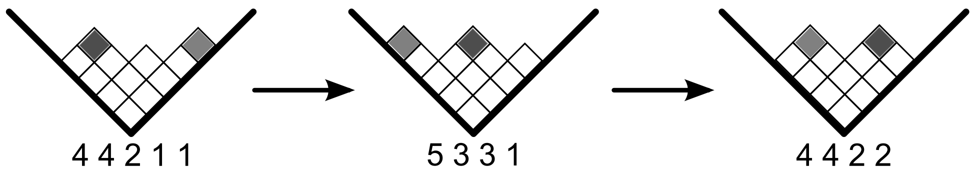

Figure 5: After convergence excessive blocks cycle on the same level.

filled, excessive blocks will cycle on the same level as shown in figure 5. □

2.5 Garden of Eden positions

A *Garden of Eden position* is a partition that can never be created during Bulgarian solitaire. The partition only appears when you start there and once it has been left, it is impossible to return. An example is shown in figure 6.

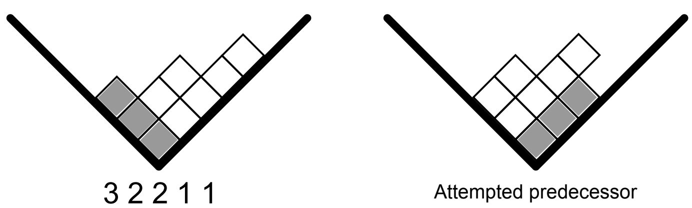

Figure 6: Because the partition has 5 terms, any predecessor must have had at least 4 terms. If the predecessor had atleast 4 terms, there must now be a pile of atleast size 4. Since the largest pile is smaller than 4, none of the piles could have been created.

Lemma 1. $\lambda_{new} \geq |\lambda| - 1$.

Any pile created in the last operation has to be greater than or equal to $|\lambda| - 1$ *(the current number of piles*-1*).*

Proof. By definition, $\lambda_{new}^{i+1} = |\lambda^i|$ and $|\lambda^i| + 1 - m = |\lambda^{i+1}|$, where m is the number of 1-piles in λ^i.

Elementary algebra shows that

$$
\begin{aligned}
|\lambda^i| + 1 - m &= |\lambda^{i+1}| \\
\lambda^{i+1}_{new} + 1 - m &= |\lambda^{i+1}| \\
\lambda^{i+1}_{new} &= |\lambda^{i+1}| - 1 + m \\
\lambda^{i+1}_{new} &\geq |\lambda^{i+1}| - 1, \text{ since } m \geq 0.
\end{aligned}
$$

□

In words, by definition the newest pile has the same size as the number of terms in the previous partition. By definition, with each operation a new pile is always formed and a number of 1-piles, m, are removed.

Since the number of terms can never increase by more than 1 per round, we know that the new term has to be atleast the current number of piles -1.

Lemma 2. *If $\lambda_1 \geq |\lambda| - 1$ the partition has a predecessor.*

Proof. One possible predecessor is always where the blocks of the largest pile were distributed over the other piles and any of its excessive blocks formed piles of size 1. Remember that λ_1 denotes (one of) the largest pile(s). If $\lambda_1 \geq |\lambda| - 1$, there exists a λ^{i-1} such that

$$\lambda^{i-1} = (\lambda^i_2 + 1, \lambda^i_3 + 1, \dots, \lambda^i_j + 1, 1, 1, \dots), \text{ where } |\lambda^{i-1}| = \lambda^i_1.$$

□

Theorem 3. $\lambda_1 < |\lambda| - 1 \Leftrightarrow$ *a Garden of Eden position.*

Proof. Lemma 1 states that any pile that was just created must be greater than or equal to $|\lambda| - 1$. This implies that if the largest pile is smaller than $|\lambda| - 1$, none of the piles could have been created. Such a partition is of course a Garden of Eden position.

$$\lambda_1 < |\lambda| - 1 \Rightarrow \text{a Garden of Eden position.}$$

Lemma 2 states that if the largest pile is larger than or equal to $|\lambda| - 1$, there is a predecessor. This is obviously not a Garden of Eden position.

$$\lambda_1 \geq |\lambda| - 1 \Rightarrow \text{not a Garden of Eden position.}$$

Since the two cases mentioned above cover all partitions, we know that the first covers all Garden of Eden positions.

$$\lambda_1 < |\lambda| - 1 \Leftrightarrow \text{a Garden of Eden position.} \qquad \square$$

3 Dual Bulgarian solitaire

3.1 Definition

In the dual version of Bulgarian solitaire, the largest pile is removed and its blocks distributed to the remaining piles one by one. The blocks are distributed from left to right (larger to smaller), with any excessive blocks forming piles of size 1. As far as we know, this game has neither been defined nor studied before.

3.2 Convergence

Theorem 4. *The dual version of Bulgarian Solitaire will converge in the same way as the original, but the partitions of a cycle appear in reverse order.*

Proof. With the cradle model, we can see that the dual version is equivalent to the original. The only difference is that the blocks move from right to left as shown in figure 7. Hence, we can safely apply both gravity and falling blocks to this cradle as well. As for the original, any

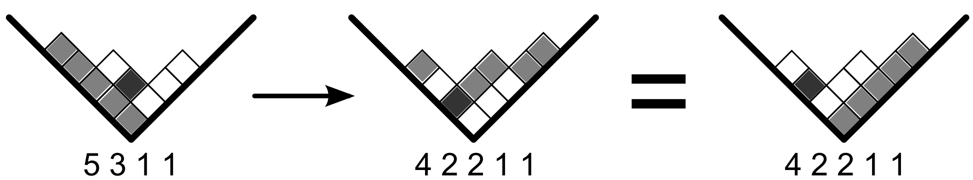

Figure 7: An operation in Dual Bulgarian solitaire can be interpreted as moving all blocks one step to the left in a cradle.

hole below a block will eventually be filled. This will result in the exact same convergence as the original, only that the partitions will appear in reverse order after convergence since surplus blocks will move from right to left. □

3.3 Garden of Eden positions

Theorem 5. *For the dual version, $|\lambda| < \lambda_1 - 1 \Leftrightarrow$ a Garden of Eden position.*

Proof. Given that there was a predecessor, the currently largest pile must have been second largest and had 1 less block in the previous partition.

$$\lambda_2^i = \lambda_1^{i+1} - 1. \tag{1}$$

By definition $\lambda_1^i \geq \lambda_2^i$, which together with (1) imply that

$$\lambda_1^i \geq \lambda_1^{i+1} - 1. \tag{2}$$

Since the largest pile of the predecessor was distributed, the current number of piles must be atleast the size of that largest pile.

$$|\lambda^{i+1}| \geq \lambda_1^i. \tag{3}$$

(2) and (3) together imply that

$$|\lambda^{i+1}| \geq \lambda_1^{i+1} - 1.$$

Hence, if $|\lambda| < \lambda_1 - 1$ our initial assumption that there was a predecessor must be false.

If $|\lambda| \geq \lambda_1 - 1$, there is a predecessor. Such a predecessor could be recreated by simply removing one block from each pile to form a new pile.

Since the two mentioned cases cover all partitions, the first covers all partitions with a predecessor so that

$$|\lambda| < \lambda_1 - 1 \Leftrightarrow \text{a Garden of Eden position.}$$ □

In the standard game, we saw that a Garden of Eden position had the properties $\lambda_1 < |\lambda| - 1$, but in the dual game it had the properties $|\lambda| + 1 < \lambda_1$. Since $|\lambda| + 1 < \lambda_1 < |\lambda| - 1$ is a contradiction, we can draw the conclusion that no partition is a Garden of Eden position in both the standard and dual game. We can also see that a partition has a predecessor in both games when $-1 \leq \lambda_1 - |\lambda| \leq 1$.

4 Multiplayer Bulgarian solitaire

4.1 Definition

In multiplayer Bulgarian solitaire several (non-empty) partitions are operated on simultaneously. Each created term is repeatedly given to another partition and no two created terms are given to the same partition.

This can be interpreted as several players sitting around a circular table playing Bulgarian solitaire. All players remove one card from each of their piles at the same time and then pass this new pile to the player on their right.

With a slight modification, the cradle model can describe the multiplayer version as well. If we let each player's piles be represented by a cradle, all blocks will move one step to the right per operation, as shown in figure 8. However, when a block cannot move further to the right within a cradle, it jumps to the next player's cradle. Note that the number of blocks in a cradle can change and that a block can fall down into a hole at the beginning of any cradle.

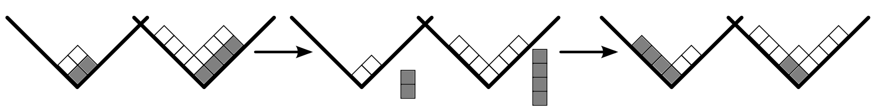

Figure 8: The figure illustrates a move in multiplayer Bulgarian solitaire for two players.

4.2 Terminology and notation

The *level* is the distance from the bottom of the cradle measured in diagonals, showed in figure 9.

p denotes the number of players.

${}_j\lambda$ denotes the partition belonging to the jth player. If j is outside of the domain, $1 \leq j \leq p$, it is defined as the number within the domain congruent to j modulo p. As an example, ${}_{p+1}\lambda$ belongs to the first player.

T_n is the largest triangular number of cards that all players can obtain simultaneously.

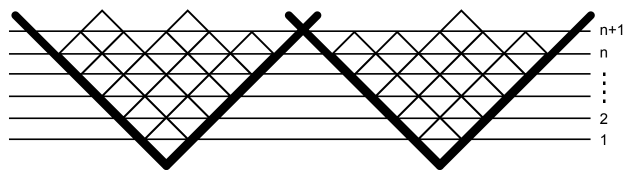

Figure 9: The figure shows the levels 1 to n+1.

4.3 Convergence

Theorem 6. *If the game has $p \cdot T_n$ cards, the game will converge so that each player has the $1, 2, 3, \ldots, n$ partition. This is obviously a stable distribution of cards.*

Proof. In the cradle model, an operation causes each block and hole to move one step to the right. If the stable distribution of blocks has not been reached, there will be blocks on higher levels than n and holes on lower levels than $n+1$. A block will fall down if it is situated directly above a hole and the distance between an extra block and a hole one level below it will decrease by one position every time the hole moves to a new cradle. This means that the distance between a hole one level below a block will eventually become 0 when the hole jumps to a new cradle so that the block falls. When there are no holes on a level below a block, each player will have a triangular partition. □

Corollary 1. *If the game has $p \cdot T_n + r < p \cdot T_{n+1}$ cards, the game will converge to so that each player has a triangular partition with r extra blocks floating on top.*

4.4 Cycle lengths

With 2 players and 8 cards, there is a cycle of length 6,

$$\underline{31 \,\&\, 22} \to 22 \,\&\, 211 \to 311 \,\&\, 21 \to 22 \,\&\, 31 \to 211 \,\&\, 22 \to 21 \,\&\, 311 \to \underline{31 \,\&\, 22}.$$

For the same number of cards and players, there is another cycle of length 3,

$$\underline{22 \,\&\, 22} \to 211 \,\&\, 211 \to 31 \,\&\, 31 \to \underline{22 \,\&\, 22}.$$

We can notice certain curiousities. In the first example the cycle seems to restart halfway through, but with the players having swapped partitions. In the second example both players have the same partitions throughout the cycle. There is a way to easily describe all cycle lengths for any given number of players and cards.

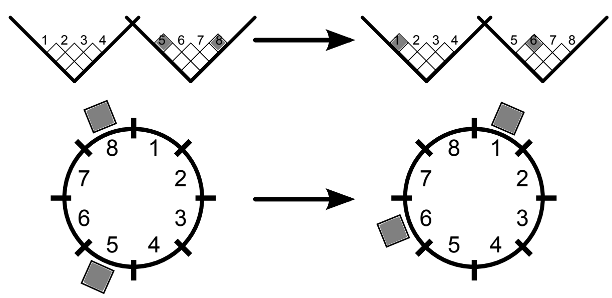

Figure 10: The surplus blocks can be regarded as moving on a circle.

After convergence a number of extra blocks, r, move along a surface of length $p(n+1)$. The last position of this surface leads to the first, which makes it more natural to think of the surface as a circle, shown in figure 10. To determine all possible cycle lengths is equivalent to determining how long time it can take for r blocks to move around a circle of length $p(n+1)$ before the same positioning reappears.

Theorem 7. *For all d that divide both r and $p(n+1)$, there is a cycle of length $\frac{p(n+1)}{d}$ and all cycle lengths are given for some such d.*

Proof. A cycle is defined as follows: After c moves, all positions previously occupied by a block will again be occupied by a block. This means that the circle has to be divisible into equal circle sectors of length c as shown in figure 13. If d is the amount of cycle sectors, it has to be true that $d \mid p(n+1)$, since $d \cdot c = p(n+1)$.

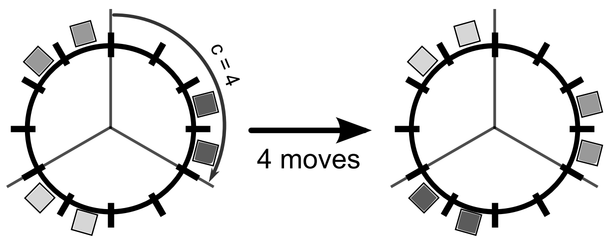

Figure 11: A cycle of length c occurs if and only if the circle is divisible into identical circle sectors of length c. After c moves, all positions occupied by a block will again be occupied by a block.

Since the circle sectors are identical, they have to include the same number of blocks. This means that r blocks have to be equally distributed into d sectors, hence $d \mid r$.

To conclude, for all cycles there is a d that divides both r and $p(n+1)$. It is obvious that all such d yields a cycle of length $c = \frac{p(n+1)}{d}$. $\square$

4.5 Number of cycles

Theorem 8.

The number of cycles $= \dfrac{1}{p(n+1)} \displaystyle\sum_{d|(p(n+1),r)} \varphi(d) \binom{p(n+1)/d}{r/d}$, *where $\varphi(d)$ is Euler's phi function.*

Proof. Determining the number of possible cycles for a given number of cards and players is difficult since several positionings of blocks belong to the same cycle. Fortunately, the problem has been solved within the field of necklace combinatorics. The number of possible cycles on a circle of length $p(n+1)$ using r blocks is the same as the number of ways a necklace of length $p(n+1)$ can be formed with r black beads and $p(n+1)-r$ white beads, as illustrated in figure 12. The problem has been solved in e.g.[2][6]. $\square$

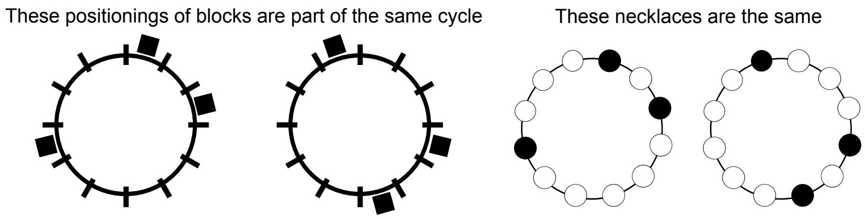

Figure 12: Determining the number of cycles is equivalent to determining the number of possible necklaces with r black beads and $p(n+1)-r$ white beads.

As an example, when there are 12 cards in play with 3 players, $n=2$ and $r=3$ so that the number of cycles is

$$\frac{1}{3(2+1)} \sum_{d|(3(2+1),3)} \varphi(d) \binom{3(2+1)/d}{3/d}$$

$$\frac{\varphi(1)\binom{3(2+1)/1}{3/1} + \varphi(3)\binom{3(2+1)/3}{3/3}}{3(2+1)}$$

$$\frac{1 \cdot \binom{9}{3} + 2 \cdot \binom{3}{1}}{9}$$

$$10.$$

4.6 Garden of Eden positions

Lemma 3. ${}_j\lambda^i_{new} \geq |{}_{j-1}\lambda^i| - 1$.

Proof. By definition, ${}_j\lambda^i_{new} = |{}_{j-1}\lambda^{i-1}|$.

By definition, a player's partition always gains one term and loses one term per pile of size 1. If m is the number of terms of size 1 in ${}_{j-1}\lambda^{i-1}$ then

$$\begin{aligned} |{}_{j-1}\lambda^{i-1}| + 1 - m &= |{}_{j-1}\lambda^{i}| \\ {}_{j}\lambda^{i}_{new} + 1 - m &= |{}_{j-1}\lambda^{i}| \\ {}_{j}\lambda^{i}_{new} &= |{}_{j-1}\lambda^{i}| - 1 + m \\ {}_{j}\lambda^{i}_{new} &\geq |{}_{j-1}\lambda^{i}| - 1, \text{ since } m \geq 0 \end{aligned}$$ □

Lemma 4. *If all players have a pile greater than or equal to $|\lambda| - 1$ for the player before him, there is a predecessor.*

Proof. If you remove the greatest pile from all players, all players will have one less pile. Since this removed pile is now greater than or equal to the number of piles that the previous player has, it could have been collected from the previous player, with its excessive size represented as piles of size 1. □

Theorem 9. ${}_{j}\lambda_1 < |{}_{j-1}\lambda| - 1 \Leftrightarrow$ *a Garden of Eden position.*

The greatest pile of at least one player is smaller than $|\lambda| - 1$ for the player before him $\Leftrightarrow$ a Garden of Eden position.

Proof. Lemma 3 $\Rightarrow$ If the greatest term any player has is smaller than $|\lambda| - 1$ for the player before him, no pile could have been created and there is no predecessor.

Lemma 4 $\Rightarrow$ If all players have a term greater than or equal to $|\lambda| - 1$ for the player before him, there is a predecessor.

Since the two mentioned cases cover all partitions, the first case covers all Garden of Eden positions so that

$${}_{j}\lambda_1 < |{}_{j-1}\lambda| - 1 \Leftrightarrow \text{a Garden of Eden position.}$$ □

5 Future research

We know that each term must have been 1 larger when (if) it was part of a predecessor. We also know that one of the current piles was not part of this predecessor. Hence, if you

All possible predecessors to 541

~~6,~~ 5, 2, 1, 1, 1
6, ~~5,~~ 2, 1, 1
6, 5, ~~2,~~ -1?

Figure 13: All possible predecessors to $(5, 4, 1)$ are $(5, 2, 1, 1, 1)$ and $(6, 2, 1, 1)$. Could we formulate rules so that $(6, 5, -1)$ could be considered a predecessor as well?

want to find a predecessor to a partition, you can simply add 1 to all terms, remove a term and add piles of size 1 until the required sum is reached. By removing each term one by one, you will find all possible predecessors. Some of these predecessors exceeds the required sum even before piles of size 1 are added, wich makes them non-real. This could be adjusted with an imaginary partition including piles of size -1. If you could interpret these imaginary partitions it might be possible to formulate a new game with interesting properties.

Another version of Bulgarian solitaire could be where only a certain percentage of the stacks are removed to form the new pile. Unfortunately the game is very predictable, it would need more rules to become interesting. Starting with a sum of m divided into piles of size $a, b, c, \ldots$ the state after k moves of removing $p \cdot 100$ percent would simply be

$$a \cdot p^k, b \cdot p^k, c \cdot p^k, \ldots, m \cdot p^k, m \cdot p^{k-1}, \ldots, m \cdot p^2, m \cdot p.$$

The obvious thing left to prove is a formula for the maximum number of moves before convergence. As far as we know, nobody has ever managed to find and prove such a formula that holds for all possible numbers of cards even for the singleplayer version.

6 Acknowledgements

First of all, I would like to thank my always so joyful mentor Henrik Eriksson for his guidance and two great weeks at KTH. I would also like to thank the people behind Research Academy for Young Scientists, namely Johannes Orstadius, Mikael Ingemyr, Nicole Nova, Fredrik Langkilde, Gustav Johansson and the xRays Isak Nilsson. Both Henrik and Rays* have spent an incredible amount of time and effort to give me this opportunity, asking for nothing in return (or well, not yet). I would like to express my greatest gratitude to the sponsors who made this research possible, primarily Institut Mittag-Leffler and Stockholms matematikcentrum but also Europaskolan for providing us with housing. Anton Grensjö, another Rays student, and I studied two different versions of Bulgarian solitaire but co-operated with the standard version. I had a great time unravelling the mysteries of Bulgarian solitaire together with him. It was Anton who took the time to write the more advanced computer simulations, especially the visualizations, and then kindly allowed me to use them as well. I am truly grateful to all who contributed to the most inspiring summer of my life.

* FOR EXCELLENCE

References

[1] Hopkins B. 30 Years of Bulgarian Solitaire. The College Mathematics Journal. 2012;43(2):135–140.

[2] Brandt J. Cycles of Partitions. Proceedings of the American Mathematical Society. 1982 Jul;85(3):483. Available from: http://www.jstor.org/stable/2043873?origin=crossref.

[3] Wolfram Alpha. Partitions of 55;. Available from: http://www.wolframalpha.com/input/?i=partitions+of+55.

[4] Eriksson H. Bulgarisk patiens. Elementa. 1981;64:186–188.

[5] Popov S. Random Bulgarian solitaire. Random Structures & Algorithms. 2008;27(3):310–330.

[6] Sawada J, Williams A. A Gray code for fixed-density necklaces and Lyndon words in constant amortized time;Available from: http://dx.doi.org/10.1016/j.tcs.2012.01.013.

A Appendix

Feel free to contact me at *emil.t.ohman@gmail.com* for the Java code written by Anton Grensjö, the Python code below or anything else.

The following code was used to simulate multiplayer Bulgarian solitaire.

```
from copy import deepcopy
def multi_operation(partitions):
   #This function performs a Bulgarian multiplayer operation on partitions
   #count the amount of players
   players = len(partitions)
   partlen = []
   #iterate over each player
   for player in range(0,players):
      #save the amount of terms the player has before we start to remove piles
      partlen.insert(player, len(partitions[player]))
      #remove 1 from all terms the player has
      for index in range(len(partitions[player]) - 1, -1, -1):
         partitions[player][index] = partitions[player][index] - 1
         #remove any term of size 0
         if(partitions[player][index]<1):
            del partitions[player][index]
   #add all players' new term and sort the terms
   for player in range(players):
      partitions[(player+1)%players].append(partlen[player])
      partitions[(player+1)%players].sort(reverse=True)
   return partitions

def comparelists(list1,list2):
   #This function checks wether two two-dimensional arrays are the same.
   if(len(list1)!=len(list2)):
      return False
   for sublist in range(len(list1)):
      if(len(list1[sublist])!=len(list2[sublist])):
         return False
      for element in range(len(list1[sublist])):
         if(list1[sublist][element]!=list2[sublist][element]):
            return False
   return True

def analyze(partitions):
   states = []
   j = 0
```

```
  while(True):
     #save a copy of all partitions in states to be able to prevent an eternal loop
     states.append(deepcopy(partitions))
     #run a round of multiplayer Bulgarian solitaire
     partitions = multi_operation(partitions)
     #j = number of Bulgarian operations
     j = j+1
     i=0
     for state in states:
        if(comparelists(state,partitions)==True):
           states.append(partitions)
           #print all states if they are fewer than 100
           if len(states)<100:
              for state in states:
                 print(state)
           print("Convergence! Time to reach loop = "+str(i)+", looplength = "+str(j-i))
           return
        #i = number of states before a match occurs
        i=i+1

example = [[4,3,2,1,1],[3,3,2,1],[4,3,2,1],[4,3,2,1]]
analyze(example)
```