



Universiteit
Leiden
The Netherlands

Nim variants

Berg, M.J.H. van den; Kusters, W.A.; Spieksma, F.M.

Citation

Berg, M. J. H. van den, Kusters, W. A., & Spieksma, F. M. (2022). Nim variants. *Icga Journal*, 44(1), 2-17. doi:10.3233/ICG-220206

Version: Publisher's Version

License: [Licensed under Article 25fa Copyright Act/Law \(Amendment Taverne\)](#)

Downloaded from: <https://hdl.handle.net/1887/4082329>

Note: To cite this publication please use the final published version (if applicable).

Nim variants

Mark van den Bergh^{a,*}, Walter Kusters^b and Flora Spieksma^a

^a *Mathematisch instituut, Universiteit Leiden, Netherlands*

^b *Leiden Institute of Advanced Computer Science, Universiteit Leiden, Netherlands*

Abstract. Combinatorial game theory provides results for the class of two-player, deterministic games with perfect information. With the aim of generalizing this theory to the class of non-perfect information games in mind, we introduce and analyze three variants of the game of Nim. In these variants, the opponent only receives partial information on the move executed by the opponent. We model the variants as games in extensive form and compute Nash equilibria for different starting configurations. For one variant, this provides a full characterization of the game. For the other variants, we prove some partial and structural results, but a full characterization remains elusive.

Keywords: Combinatorial game theory, Nim, game in extensive form, Nash equilibrium, Kriegspiel

1. INTRODUCTION

The field of combinatorial game theory considers the class of two-person, deterministic games with perfect information. For these games, a beautiful theory has been established, showcased in Berlekamp et al. (1982), Siegel (2013) and Albert et al. (2019). Games belonging to this class are, e.g., Nim, Hackenbush, Hex and Domineering. Some recent advances on the latter game can be found in Uiterwijk (2016), and a comprehensive guide to Hex has been published in Hayward and Toft (2019).

The central question we will address, is what remains of said theoretical framework if we omit the assumption of perfect information. In the case of chess, for example, which may also be considered to belong to the class of combinatorial games, by leaving the premise of perfect information, one may end up with the game of Kriegspiel. In that game, both players do not see the full state of the board, and must communicate their moves via an all-seeing umpire. Another game in this category is poker, in which a player lacks information on the cards in other players' hands.

In this paper, we introduce three variants of a non-perfect information version of the well-known impartial game Nim. As a first example, consider a simple Nim configuration, depicted in Fig. 1. In regular Nim, the winning strategy for the first player is to remove one chip from any heap.



Fig. 1. A simple Nim position with three heaps of sizes 3, 3 and 1, respectively.

The first variant we consider is *Schrödinger Nim*. The difference with regular Nim is that the opponent is only told from which heap chips have been removed, but not how many. Hence, a move now consists of first selecting a heap to inspect, and then removing any amount of chips from it. If a heap is emptied by a move, this is communicated to the opponent.

*Corresponding author. E-mail: m.j.h.van.den.bergh@math.leidenuniv.nl.

Because of the introduction of imperfect information, a strategy for a player now consists of a probability distribution over their possible moves. We assign value 1 to a position which is won by the first player with probability 1, and -1 if the game is won by the second player. The position in Fig. 1 has value $1/3$; a pair of optimal strategies is illustrated in Fig. 2. Here, “optimal” refers to a Nash equilibrium: it can be shown that the players cannot improve by one-sided deviation from their strategy.

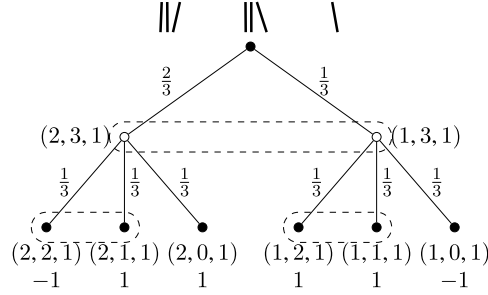


Fig. 2. Optimal strategies for Schrödinger Nim played on heaps of sizes 3, 3 and 1. The states in dotted ovals cannot be distinguished by the current player. In the leaves, final values are depicted, omitting further play.

The first player either removes 1 or 2 chips from the first heap, with probability $2/3$ and $1/3$, respectively. The second player, not knowing how many chips are still in the first heap, removes any number of chips from the second heap with equal probability. If the second player emptied the second heap, the game values are easily computed. Otherwise, if the first player removed one chip from the first heap on the first move, she later proceeds by emptying this heap, winning if and only if the second heap contains 1 chip. Finally, if the first player removed two chips on the first turn, she later proceeds to empty the second heap, winning the game.

We show that, if all heaps consist of at most two chips, Schrödinger Nim positions are equivalent to regular Nim positions. Conversely, if all heaps consist of at least three chips, Schrödinger Nim positions have value 0, i.e., they do not favor either player (see Theorem 3.1). The truly interesting positions are those in which there are some heaps of size at most two, and some of size at least three, as in the above example. We consider such positions with three heaps, and provide several results.

The second variant we consider is *Fuzzy Schrödinger Nim*, following the same rules as Schrödinger Nim, except that emptying a heap is no longer signalled to the opponent. In Theorem 4.1 and Theorem 4.2 we give a complete characterization of this game, as well as its misère variant. Finally, we consider a third version, called *Kriegspiel Nim*, for which we use a set of rules inspired by the chess variant Kriegspiel. Theorem 5.1 characterizes games with two heaps.

For both combinatorial games and games without perfect information, research has been done in the field of artificial intelligence to create powerful agents. Examples include the application of Monte-Carlo Tree Search (MCTS) and deep neural networks to Go in Silver et al. (2017), using MCTS and meta position based agents to play Kriegspiel in Ciancarini and Favini (2007, 2010), and exploring Counterfactual Regret Minimization and deep neural networks in the context of Heads-up No-limit Poker in Bowling et al. (2017) and Moravčík et al. (2017). Whereas these methods produce powerful artificial players also in the context of non-perfect information, we are interested in developing a theory akin to that for combinatorial games, for this class of games.

Research in this direction has been done in the context of synchronizing combinatorial games in Cincotti and Iida (2007) and Huggan and Nowakowski (2018). In these synchronized versions, both

players communicate an intended move to an umpire, after which these moves are executed simultaneously. Though results are promising for several partisan games, this approach might prove problematic for impartial games, as both players may have selected the same move to execute.

The introduced (Fuzzy) Schrödinger imperfect information variant of the rules of Nim can be generalized to any (impartial) game in which the amount of disjunctive components is non-increasing. Variants of Nim such as arbitrary finite subtraction games come to mind, but also a game like Push (see Albert et al. (2019)) can be altered in a natural way such that the opponent only hears the component on which a move was made, but not the move itself. The Kriegspiel rule variant can be applied to any combinatorial game.

In Section 2, we more formally outline the rules of the Nim variants, as well as their modelling as games in extensive form for computational purposes, as outlined in Ferguson (2014); Kuhn (1953); Nisan et al. (2007); Von Stengel (2002). In Section 3, Section 4 and Section 5, we analyse the three variants, finding a full characterization for Fuzzy Schrödinger Nim, and partial results for the other two. We conclude in Section 6.

2. GAME RULES AND NOTATION

In this section, we introduce the rules and notation for three variants of Nim. Furthermore, we introduce notation for games in extensive form, which will be used for the analysis of the variants.

2.1. Nim and its variants

The classic combinatorial game of Nim starts with a configuration of d heaps, the i -th heap consisting of n_i chips. Two players L and R alternate turns, each player being allowed to take any number of chips from any one heap every turn. The player taking the last chip wins. Nim being an impartial game, for the remainder of the paper, we assume without loss of generality that L starts. Denoting the height of the heaps by $n = (n_1, \dots, n_d)$, we write $\text{NIM}(n)$ for the game of Nim with these heaps as starting configuration.

In the first variant of the game, dubbed *Schrödinger Nim*, both players know the starting configuration. During the game, however, when the opponent moves, a player is only told which heap the move was on, not the amount of chips removed. An exception is when a heap is emptied, which is always communicated to the opponent. The height of (some of) the heaps may thus be unknown to a player during the game. A move now consists of first selecting a heap, at which point the height of the heap becomes known, followed by making a decision on how many chips to remove from the heap. We denote a game of Schrödinger Nim with starting configuration $n = (n_1, \dots, n_d)$ by $\text{SN}(n)$. Note that the game is impartial and short.

The second variant, which we will call *Fuzzy Schrödinger Nim*, is very much like the first, again with both players knowing the starting configuration. However, in the fuzzy version, emptying a heap is not signalled to the other player. Now, a player may happen to select an empty heap to move on. In this case, the player is told the heap is empty, and they must pick another heap to try and remove chips from. A game of this variant with starting configuration $n = (n_1, \dots, n_d)$ is denoted by $\text{FSN}(n)$. We will return to this variant in Section 4.

The third variant which we will consider is inspired by Kriegspiel and is therefore named *Kriegspiel Nim*. Yet again, both players know the starting configuration. During a player's turn, a move consists

of trying to remove i chips from the j -th heap. If this is possible, it is done, and the turn is passed, the other player not being informed of anything except the fact that a successful move has been executed. In particular, if a pile is emptied by a move, this is not explicitly communicated to either of the players. If the requested move is not possible, the player must try another move, continuing until a legal move is tried and thus performed. For $n = (n_1, \dots, n_d)$, we denote this variant of the game with starting position n by $\text{KN}(n)$.

To illustrate the three game variants, again consider the Nim position from Fig. 1. Suppose that the first player removes one chip from the first heap. In both Schrödinger and Fuzzy Schrödinger Nim, the second player only knows that the first heap has been altered, whereas in Schrödinger Nim he also knows that the heap is non-empty. The possible moves for the second player are now either to remove any number of coins from the second or third heap, or to choose to inspect the first heap. If he chooses to inspect the first heap, he discovers there are two chips left, and may then proceed to choose to remove one or both of them.

In Kriegspiel Nim, after the first player has removed a chip from the first heap, the second player receives no information at all. His move now consists of trying to remove any amount of chips from any heap. If he would try to remove three chips from the first heap, the umpire would respond negatively, revealing that the first player has removed at least one chip from the first heap. Any other attempted move that was legal from the (known) starting position can and will be executed. See Fig. 3 for an illustration of this example.

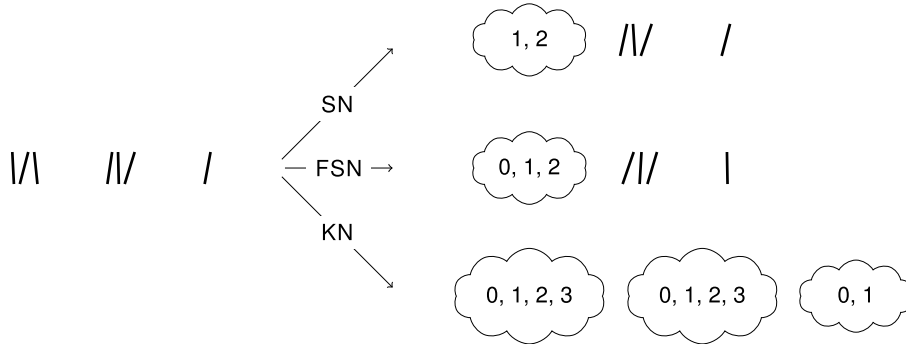


Fig. 3. The starting position of a game with three heaps of sizes 3, 3 and 1, respectively, as seen by R after L's first move, in the three different variants of Nim. The clouds contain the possible heights of the heaps. Notice that in Kriegspiel Nim, R knows a little more: exactly one of the heaps has been touched.

2.2. Extensive form games

As the players no longer have perfect information in the variants of Nim described above, the variants are no longer combinatorial games. Instead, we model them as games in extensive form as in Ferguson (2014); Kuhn (1953); Nisan et al. (2007). In such a game, we have a set of states X . The set $T \subseteq X$ is the set of terminal states, each having value $v(x) \in \mathbb{R}$, $x \in T$, and the non-terminal states are partitioned into (disjoint) information sets. For every information set S , we define an action set $A(S)$. Every action $a \in A(S)$ made in a state $x \in S$ leads to a unique state $\tau(x, a)$. We write $X(x)$ for all states reachable by one action from x . We denote the last action made to reach state x by $\alpha(x)$. An action sequence is a list of consecutive actions, starting from the initial state: $\sigma = [a_1, \dots, a_n]$. The unique state to which this sequence leads is denoted by $\tau(\sigma)$. The part of σ consisting of only L's actions is denoted by σ_L ; the definition of σ_R is analogous.

A common assumption is that of *perfect recall*: no two different action sequences by one player can lead to the same information set. That is, if for two sequences σ, σ' and a player $P \in \{L, R\}$ we have $\sigma_P \neq \sigma'_P, \tau(\sigma) \in S$ and $\tau(\sigma') \in S'$, then $S \neq S'$. Under this assumption, we can denote a player's unique action sequence leading to an information set $S \in \mathcal{S}$ by $\sigma_P(S)$. Note that by recording the sequence leading to an information set in the description of the set, i.e., by assuming that both players do not forget earlier moves they did, we can always guarantee perfect recall.

A (*behavior*) strategy π defines on every action set a probability distribution. The probability of choosing any $a \in A(S)$ for $S \in \mathcal{S}$ is denoted by $\pi(a)$. The value of a strategy pair (π_L, π_R) for the two players L and R is denoted by $v(\pi_L, \pi_R)$. The value of a strategy pair when starting the game in a state $x \in X$ is denoted by $v_x(\pi)$.

For the Nim variants, we take as states the current height of the heaps, and the current player to move. The terminal states are those with only empty heaps, having value 1 if it is R's turn (so L, the starting player, wins) and -1 otherwise, giving rise to a zero-sum game. For the division of the rest of the states into information sets, we keep track of the current player, the action sequence for that player so far, and the knowledge of the player of the height of the heaps. For every heap k , this can either be an exact height, say h , or an upper bound, denoted by \bar{h} , if the other player has moved on the heap last.

For the games of (Fuzzy) Schrödinger Nim, the current player can either choose to remove any amount of chips from a known heap, or to take a look at an unknown heap. In the latter case, the player sees the amount of chips left, entering another information state, from which they can only choose to remove any amount of chips from the chosen heap. We denote the move of removing i chips from the (known) k -th heap by k_i .

3. SCHRÖDINGER NIM

In this section, we analyze the game of Schrödinger Nim. We first argue that configurations in which either all heaps are sufficiently small or all heaps are sufficiently large, the outcome is straightforward. Then, we continue by partially analyzing the game on three heaps of varying size, which proves to be more complicated.

3.1. Basic cases

First, note that if the other player has moved on a heap i times since the first player has observed its height to be h , their knowledge of the heap will be $\bar{h} - i$. Furthermore, we have that $\bar{1}$ is equivalent to 1, as the emptying of a heap is signalled. Therefore, if a heap contains two chips, and both players know, the height of that heap will be known to both players for the rest of the game. Hence, $\text{SN}(n)$ is equivalent to $\text{NIM}(n)$ if $n_i \leq 2$ for all $i = 1, \dots, d$. Naturally, in the trivial case that the game starts with only one heap, Schrödinger Nim is also equivalent to regular Nim. Finally, if all remaining heaps have height 1 at some point in the game, regardless of the information both players have, the result will be the same as for regular Nim, and is determined by the parity of the number of non-empty heaps.

In these cases, both players can use a deterministic strategy when playing optimally, and the starting player L will either always win or always lose if both players play optimally. In other cases, the game may not be always won or lost by the starting player. Instead, the players will employ a Nash equilibrium of strategies, having a real value $v \in [-1, 1]$, where $\frac{v+1}{2}$ is the probability of L winning.

We denote by $v(\text{SN}(n))$ the value of the game, which is the unique value of some Nash equilibrium for the two players.

Theorem 3.1. *Let $n \in \mathbb{N}_{\geq 3}^d$, $d \geq 2$. Then $v(\text{SN}(n)) = 0$.*

Proof. We proceed by induction on d . For the base case $d = 2$, we provide an explicit strategy pair and prove that it is Nash. On the first turn, L reduces one of the heaps to one or two chips, both with probability $\frac{1}{2}$. Without loss of generality, let this be the first heap. Next, R does the same for the other heap. Now, if L reduced her heap to two chips, she takes one of the remaining chips there. Otherwise, she takes one chip from the other heap. R again mirrors. Note that the value of the strategy pair is indeed 0.

The equilibrium is depicted in Fig. 4. The nodes are states, the dotted boxes show information sets. Note that in fact every node is contained in an information set, but only the most relevant sets are displayed. By the discussion above, we may interpret states in which both heaps consist of one chip as terminal states, yielding a value of either 1 or -1 .

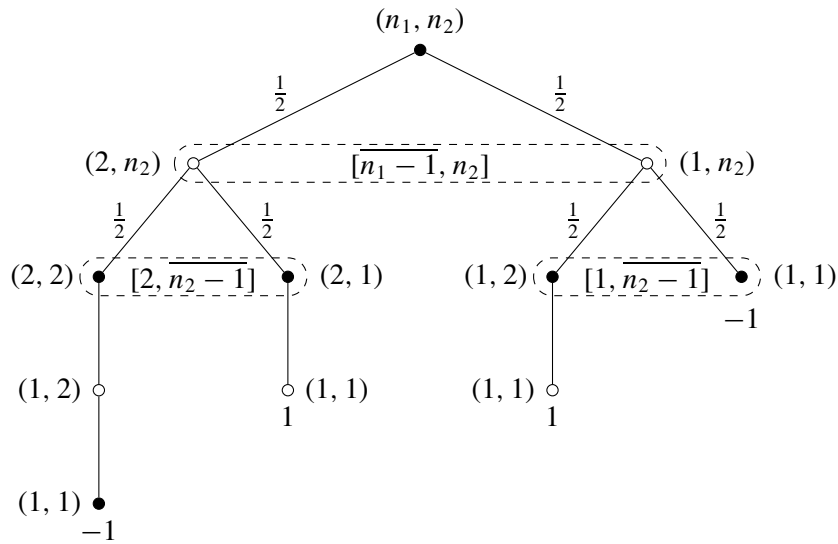


Fig. 4. The Nash equilibrium in action. In black nodes, it is L's turn; in white nodes, it is R's turn. The action probabilities are shown on the edges. The labels of the information sets display the information as known to the active player.

The description of the strategies above is not exhaustive. Therefore, in the sequel, if we encounter an information set in which no probability distribution over the actions has been defined yet, we must and will do so, making sure that this does not contradict an earlier definition. For example, it has not been decided how Left should play in the information set $[1, n_2]$, which may be reached by Right playing on the first heap during his first turn. However, it is clear that, in this case, 2_{n_2-1} is an optimal move, which we would have filled in only when encountering this information set for the first time.

We prove that the provided strategy pair is indeed Nash. First, suppose R deviates. If R checks and plays on the first heap during the first move, L is given the state $(1, n_2)$ or $(0, n_2)$ with perfect knowledge, winning the game by removing all but one or all chips from the second heap, respectively. Similarly, emptying the second heap leads to a win for L. Therefore, we may suppose that R plays 2_i with probability p_i , with $\sum_{i=1}^{n_2-1} p_i = 1$, $p_{n_2} = 0$.

If R was in $(2, n_2)$, L will respond by moving to $(1, n_2 - i)$, which is a win for R if and only if $n_2 - i \neq 1$. If R was in $(1, n_2)$, L will inspect and play on the second heap, which is a win for R if and only if $n_2 - i = 1$. Both cases occurring with probability $\frac{1}{2}$, we find that R obtains a value of 0 regardless of the p_i . Noting that it is clear that R cannot gain by deviating during his second turn, we conclude that R cannot improve his score by deviating at all.

Next, suppose L deviates. Emptying a heap leads to a win for R, so assume that L plays $\mathbf{1}_i$ with probability p_i , where $\sum_{i=1}^{n_1-1} p_i = 1$, $p_{n_1} = 0$. Next, R moves to either $(n_1 - i, 2)$ or $(n_1 - i, 1)$ with probability $\frac{1}{2}$. If $n_1 - i = 1$, L must play on the second heap to not immediately lose, winning if and only if R left 2 chips. Otherwise, playing on the second heap allows R to certainly win, so suppose L plays $\mathbf{1}_j$ with probability q_j , where $\sum_{j=1}^{n_1-i-1} q_j = 1$, q_{n_1-i} . If R initially left 2 chips, he moves to $(n_1 - i - j, 1)$, which is winning for L if and only if $n_1 - i - j \neq 1$. Otherwise, he inspects and plays on the first heap, which is winning for L if and only if $n_1 - i - j = 1$. Hence, L obtains value 0 regardless of the p_i and q_j and cannot gain by deviating. We conclude that the given strategy pair is Nash.

Finally, suppose $v(\text{SN}(n)) = 0$ for any $n \in \mathbb{N}_{\geq 3}^d$ and consider the game $\text{SN}(n)$ with $n \in \mathbb{N}_{\geq 3}^{d+1}$. If L empties any heap in the first move, this guarantees a value of 0. Otherwise, R is faced with d heaps of known height and one of unknown height. Now, by checking and emptying the heap of unknown height, R can again guarantee a value of 0. By playing on any other heap, he might force a value of less than 0. Therefore, the best L can do is to prevent this by emptying any heap on the first move, obtaining value 0. \square

The above theorem shows that either player wins with probability $\frac{1}{2}$ if all heaps contain at least 3 chips at the start of the game. For the game with two heaps, if either heap is initially of height 1 or 2, L can win by reducing the other heap to the same size. Games with three or more heaps, in which some heaps have at most 2 chips and some at least 3, however, turn out to be more difficult to analyze, as we will see in the next subsection.

3.2. Three heaps

In this section, we give a partial characterization of $v(\text{SN}(n))$ for $n \in \mathbb{N}^3$. The values for all starting configurations except for $(n_1, n_2, 1)$ with $n_1, n_2 \geq 4$ are given in Table 1. These values have been determined as follows. First, we computed Nash equilibria and their values for some small cases of the game using the sequence form and linear programming, see, e.g., (Nisan et al., 2007, Chapter 3). Consequently, having found structure in the equilibria, we constructed a strategy pair for the general case and proved that it is Nash similar to the proof of Theorem 3.1. Note that the heaps may be permuted in any way without changing the value of the game. The game $(1, n_1, 1)$, for example, also has value 1 for all $n_1 \geq 3$.

Table 1
Values of three-heap games

$(n_1, 1, 1)$ $n_1 \geq 1$	$(3, 2, 1)$	$(n_1, 2, 1)$ $n_1 \geq 4$	$(n_1, 3, 1)$ $n_1 \geq 3$	$(n_1, 2, 2)$ $n_1 \geq 3$	$(n_1, n_2, 2)$ $n_1, n_2 \geq 3$	(n_1, n_2, n_3) $n_1, n_2, n_3 \geq 3$
1	0	1/2	1/3	1	1/3	0

The games $\text{SN}(n_1, n_2, 1)$ with $n_1, n_2 \geq 4$ do not have a constant value. Some computed values for small n_1 and n_2 are shown in Table 2.

Table 2
Values for the game $\text{SN}(n_1, n_2, 1)$

$n_2 \setminus n_1$	4	5	6	7
4	1/5	1/4	1/4	1/4
5		3/13	5/21	5/21
6			4/17	13/55

For this configuration of the game, we present some structural results. The proofs of these results (partly) rely on the following observation, stating the principle of indifference for games in extensive form. Note that this is true for these games in general, not only for Schrödinger Nim.

Theorem 3.2. *Let $\pi = (\pi_L, \pi_R)$ be a Nash equilibrium for a two-player zero-sum game in extensive form with perfect recall, in which L moves first and R moves second. Write $v = v(\pi)$, and let r be the root of the Kuhn tree of the game. Then*

- (1) $v_x(\pi) = v$ for all $x \in X(r)$ with $\pi_L(\alpha(x)) > 0$.
- (2) $\sum_{x \in S} \pi_L(\alpha(x)) v_{\tau(x,a)}(\pi) = \sum_{x \in S} \pi_L(\alpha(x)) v$ for all $a \in A(S)$ with $\pi_R(a) > 0$, where S is an information set for which $\sigma_R(S) = \emptyset$.

The proof is relatively straightforward and consists of showing that, if a move made with positive probability would yield any other value than the value of the game, it would be beneficial to play this move with a different lower probability, contradicting the fact that the players follow an equilibrium strategy.

Lemma 3.1. *Let $n_2 \in \mathbb{N}$, $n_2 \geq 3$. Then $v(\text{SN}(n_1, n_2, 1))$ is non-decreasing in n_1 , for $n_1 \geq n_2$.*

Proof. Consider the game $\text{SN}(n_2, n_2, 1)$. Suppose first that there exists a Nash equilibrium (π_L, π_R) for which $\pi_L(\mathbf{3}_1) = 0$ with payoff $v := v(\text{SN}(n_2, n_2, 1))$. Then, for the game $\text{SN}(n_1, n_2, 1)$ with $n_1 > n_2$, L may use the same strategy as for the game $\text{SN}(n_2, n_2, 1)$, playing $\mathbf{1}_i$ with $i > n_1 - n_2$ on the first turn to obtain a value of v in this game, as well.

If such an equilibrium does not exist, let $\pi = (\pi_L, \pi_R)$ be Nash such that $\pi_L(\mathbf{3}_1) > 0$. Note that $0 = v(\text{SN}(n_2, n_2, 0)) = v_{\mathbf{3}_1}(\pi) = v(\text{SN}(n_2, n_2, 1))$ by Theorem 3.2. Therefore, we may define $\pi' = (\pi'_L, \pi_R)$ with $\pi'_L(\mathbf{3}_1) = 1$ as another Nash equilibrium strategy pair. Now, for $\text{SN}(n_1, n_2, 1)$ with $n_1 > n_2$, L may also force a value of 0 by picking $\mathbf{3}_1$ with probability 1 as first move. Hence, indeed $v(\text{SN}(n_1, n_2, 1)) \geq v(\text{SN}(n_2, n_2, 1))$ for $n_1 > n_2$. \square

Lemma 3.2. *Let $n_1, n_2 \in \mathbb{N}$, $n_1, n_2 \geq 4$. Then $v(\text{SN}(n_1, n_2, 1)) \in [0, \frac{1}{4}]$.*

Proof. By playing $\mathbf{3}_1$ as first move, L guarantees a value of 0 by Theorem 3.1. Therefore, $v(\text{SN}(n_1, n_2, 1)) \geq 0$.

To prove the upper bound of $\frac{1}{4}$, we give an explicit strategy for R against which L cannot obtain a value of more than $\frac{1}{4}$. Suppose without loss of generality that L plays on the first heap during her first move. Emptying this heap leads to a win for R, so we may assume that L plays $\mathbf{1}_i$ for some $i \in \{1, \dots, n_1 - 1\}$.

Now, let R play $\mathbf{2}_{n_2-3}$ with probability $\frac{1}{8}$, $\mathbf{2}_{n_2-2}$ or $\mathbf{2}_{n_2-1}$ both with probability $\frac{2}{8}$ and $\mathbf{2}_{n_2}$ with probability $\frac{3}{8}$. If L played $\mathbf{1}_{n_1-1}$, this yields a value of $\frac{1}{8} + \frac{2}{8} + \frac{2}{8} - \frac{3}{8} = \frac{2}{8}$. If L played $\mathbf{1}_i$ for any

$i \in \{1, \dots, n_1 - 2\}$, R's move $\mathbf{2}_{n_2}$ leads to a win for L, whereas any of his other moves leads to the information set $[\overline{n_1 - i}, \overline{n_2 - 1}, 1]$ for L. We continue by showing that L cannot obtain more than a value of $-\frac{1}{8}$ in any of these information sets, so the total value obtained will be at most $-\frac{1}{8} + \frac{3}{8} = \frac{2}{8}$.

First, note that in these information sets, inspecting the second heap will lead to a loss for L if it was of height 2 or 1, which is the case with probability $\frac{4}{8}$. Hence, this move will yield a value of at most $\frac{1}{8} - \frac{4}{8} = -\frac{3}{8} < -\frac{1}{8}$. Similarly, playing $\mathbf{1}_{n_1-i}$ yields a win for L if and only if the height of the second heap was 1, so this move results in a value of $-\frac{1}{8} - \frac{2}{8} + \frac{2}{8} = -\frac{1}{8}$. We may thus assume that L plays $\mathbf{1}_j$ with probability q_j for $j \in \{1, \dots, n_1 - i - 1\}$, facing R with the information set $[\overline{n_1 - 2}, \ell, 1]$ with $\ell \in \{1, 2, 3\}$; or that L plays $\mathbf{3}_1$ with probability r , leading to $[\overline{n_1 - 1}, \ell, 0]$ with $\ell \in \{1, 2, 3\}$.

In $[\overline{n_1 - 2}, 3, 1]$, let R play $\mathbf{2}_1$. In $[\overline{n_1 - 2}, 2, 1]$, R plays $\mathbf{2}_1$ or $\mathbf{2}_2$ both with probability $\frac{1}{2}$. This leaves L either in $[\overline{n_1 - i - j}, \overline{n_2 - 2}, 1]$ or $[\overline{n_1 - i - j}, 0, 1]$. In the latter case, L wins if and only if $n_1 - i - j \neq 1$. In the former case, choosing to empty the first heap, say with probability s , leads to a win for L if the second heap consists of one chip, and a loss if it consists of two. Doing anything else leads to a loss for L if the second heap contains one chip. Finally, in $[\overline{n_1 - 2}, 1, 1]$, R inspects and empties the first heap, which leads to a loss for L.

In $[\overline{n_1 - 1}, 3, 0]$, let R play $\mathbf{2}_1$. Consequently, R will play as if he leaves L with two heaps of height 2, winning if and only if this is the case. In $[\overline{n_1 - 1}, 2, 0]$, R inspects the first heap, and reduces it to height 2 if possible, winning if this is the case, and losing otherwise. In $[\overline{n_1 - 1}, 1, 0]$, R also inspects the first heap, and consequently reduces it to one chip, winning the game.

Altogether, starting from the information set $[\overline{n_1 - i}, \overline{n_2 - 1}, 1]$ with $n_1 - i \geq 3$, L will thus obtain a value of at most

$$\begin{aligned} & \frac{1}{8} \left(\sum_{j=1}^{n_1-i-2} q_j ((1-s) - s) + q_{n_1-i-1} + r \right) \\ & + \frac{2}{8} \left(\sum_{j=1}^{n_1-i-2} q_j \left(\frac{1}{2} (-(1-s) + s) + \frac{1}{2} \right) + q_{n_1-i-1} \left(\frac{1}{2} - \frac{1}{2} \right) - r \right) - \frac{2}{8} \\ & = \frac{1}{8} \sum_{j=1}^{n_1-i-1} q_j - \frac{1}{8} r - \frac{2}{8} \leq -\frac{1}{8}. \end{aligned}$$

If $n_1 - i = 2$, the signs in front of the r 's are flipped, also leading to a total of at most $-\frac{1}{8}$. Hence, we indeed see that L indeed cannot obtain a value of more than $\frac{2}{8}$. \square

Corollary 3.1. *Let $n_1 \in \mathbb{N}$, $n_1 \geq 5$. Then $v(\text{SN}(n_1, 4, 1)) = \frac{1}{4}$.*

Proof. We compute that $v(\text{SN}(5, 4, 1)) = \frac{1}{4}$ using a sequence form linear programming formulation of the game. The result then immediately follows from Lemma 3.1 and Lemma 3.2. \square

Corollary 3.2. *Let $n_1, n_2 \in \mathbb{N}$, $n_1, n_2 \geq 3$, $n_1 \geq n_2$. Then there exists a Nash equilibrium $\pi = (\pi_L, \pi_R)$ for which*

$$1 - 2\pi_L(\mathbf{1}_{n_1-1}) = 1 - 2\pi_R(\mathbf{2}_{n_2})$$

for the first turns of L and R . Moreover, if $\pi_L(\mathbf{1}_{n_1-1}) > 0$, it holds that

$$v(\text{SN}(n_1, n_2, 1)) = 1 - 2\pi_L(\mathbf{1}_{n_1-1}).$$

Proof. If $\pi_L(\mathbf{1}_{n_1-1}) = \pi_R(\mathbf{2}_{n_2}) = 0$, the statement is trivial. If $v(\text{SN}(n_1, n_2, 1)) = 0$, there exists an equilibrium for which this holds, with $\pi_L(\mathbf{3}_1) = 1$. Therefore, we may assume for the remainder of the proof that $v(\text{SN}(n_1, n_2, 1)) > 0$ and $\pi_L(\mathbf{3}_1) = 0$.

Next, by Theorem 3.2, if $\pi_L(\mathbf{1}_i) > 0$ for some $i \in \{1, \dots, n_1 - 1\}$, as all these moves lead to the information set $[\overline{n_1 - 1}, n_2, 1]$ for Right, there exists a Nash equilibrium for which $\sum_{i=1}^{n_1-1} \pi_L(\mathbf{1}_i) = 1$.

If $\pi_L(\mathbf{1}_{n_1-1}) > 0$, by Theorem 3.2, $v = v_{(1, n_2, 1)}(\pi)$ with R moving. Noting that R wins if and only if he removes all chips from the second heap, we find

$$v = v_\pi(1, n_2, 1) = -\pi_R(\mathbf{2}_{n_2}) + (1 - \pi_R(\mathbf{2}_{n_2})) = 1 - 2\pi_R(\mathbf{2}_{n_2}).$$

Conversely, if $\pi_R(\mathbf{2}_{n_2}) > 0$, again by Theorem 3.2, we find

$$\sum_{i=1}^{n_1-1} \pi_L(\mathbf{1}_i)v = \sum_{i=1}^{n_1-1} \pi_L(\mathbf{1}_i)v_{(i, 0, 1)}(\pi) = \sum_{i=1}^{n_1-2} \pi_L(\mathbf{1}_i) - \pi_L(\mathbf{1}_{n_1-1}).$$

If also $\pi_L(\mathbf{1}_{n_1-1}) > 0$, then we may assume that $\sum_{i=1}^{n_1-1} \pi_L(\mathbf{1}_i) = 1$ so that $v = 1 - 2\pi_L(\mathbf{1}_{n_1-1})$. Hence, if both $\pi_L(\mathbf{1}_{n_1-1}) > 0$ and $\pi_R(\mathbf{2}_{n_2}) > 0$, we are done.

Therefore, suppose that $\pi_L(\mathbf{1}_{n_1-1}) > 0$, but $\pi_R(\mathbf{2}_{n_2}) = 0$. Then $v = 1$. However, this is a contradiction with Lemma 3.2 and the values for $\text{SN}(n_1, n_2, 1)$ with $n_1 = 3$ or $n_2 = 3$.

Finally, suppose that $\pi_R(\mathbf{2}_{n_2}) > 0$, but $\pi_L(\mathbf{1}_{n_1-1}) = 0$. Then $\sum_{i=1}^{n_1-2} \pi_L(\mathbf{1}_i)v = \sum_{i=1}^{n_1-2} \pi_L(\mathbf{1}_i)$, so again $v = 1$, which is a contradiction. This completes the argument. \square

4. FUZZY SCHRÖDINGER NIM

In this section, we consider the variant of the game in which the emptying of a heap is not signalled to the other player. Now, it may happen that a player inspects the height of a heap only to find out that it is empty. In this case, the player must choose another heap to perform a move on, continuing until the player has successfully removed at least one chip. Recall that a game of this variant with starting configuration $n \in \mathbb{N}^d$ is denoted by $\text{FSN}(n)$.

We have the following complete characterization.

Theorem 4.1. *Let $n \in \mathbb{N}^d$. Then*

$$v(\text{FSN}(n)) = \begin{cases} -1 & \text{if } \max_{k=1, \dots, d} n_k = 1 \text{ and } d \text{ is even,} \\ 1 & \text{if } \max_{k=1, \dots, d} n_k = 1 \text{ and } d \text{ is odd,} \\ & \text{or if } n_k > 1 \text{ for precisely one } k, \\ 0 & \text{otherwise.} \end{cases}$$

Proof. If we have $\max_{k=1,\dots,n} n_k = 1$, the game is equivalent to regular Nim, so L wins if and only if there is an odd number of chips available. If $n_k > 1$ for precisely one k , L can reduce this heap to either 1 or 0 chips, depending on the parity of d , winning the game. This settles the first two cases.

For the last case, we distinguish between two situations. First, suppose that $n_k \geq 2$ for all $k = 1, \dots, d$. The proof for this situation is analogous to that of Theorem 3.1, employing induction on d . For the base case $d = 2$, we use a similar Nash strategy pair. Now, L reduces the first heap to size 0 or 1 both with probability $\frac{1}{2}$, and R does the same to the second heap. L wins precisely if one of the heaps has size 1 after the initial moves, which happens with probability $\frac{1}{2}$, hence the value is indeed 0. Proving that the pair is indeed Nash is analogous to the proof in Theorem 3.1.

For the induction step, suppose $v(\text{FSN}(n)) = 0$ for all $n \in \mathbb{N}$ as above and consider the game $\text{FSN}(n)$ for some $n \in \mathbb{N}^{d+1}$ with $n_k \geq 2$ for all k . Without loss of generality, let L play on the first heap on her first turn. Now, by also playing on the first heap during his first turn, R can guarantee a value of at most 0. Indeed, if the heap was already emptied by L, R will find out and be in the case $n \in \mathbb{N}^d$. Otherwise, R can empty the heap himself. Therefore, the best L can do is to empty any heap, giving value 0. This concludes the first situation.

For the second situation, in which at least one of the n_k equals 1, we split the proof into three parts. First, consider the game $\text{FSN}(n_1, n_2, 1)$ with $n_1, n_2 \geq 2$. We will show that the value of this game is 0. First, note that emptying the third heap on the first move of L leads to a value of 0. Remains to show that L cannot achieve a payoff of more than 0 by choosing another strategy. Hence, without loss of generality, let L pick i chips from the first heap with probability p_i . We give an explicit strategy for R which prohibits L from obtaining a payoff larger than 0.

On his first move, R is faced with the information set $[\overline{n_1 - 1}, n_2, 1]$. With probability $\frac{1}{2}$, R reduces the second heap to height 1, and with equal probability, R empties the heap. L is thus given the info set $[n_1 - i, \overline{n_2 - 1}, 1]$.

For the case $n_1 - i = 1$, this yields either the state $(1, 1, 1)$ or $(1, 0, 1)$, which gives value 1 or -1 , respectively. For $n_1 - i = 0$, it results in the state $(0, 1, 1)$ or $(0, 0, 1)$, giving value -1 or 1, respectively.

Now, suppose $n_1 - i > 1$, and let L take j chips from the first heap with probability $q_j^{n_1-i}$, $j \in \{1, \dots, n_1 - i\}$. L inspects the second heap with probability $r_1^{n_1-i}$ and L empties the third heap with probability $r_2^{n_1-i}$. For $j = 1, \dots, n_1 - i - 2$, R ends up in the state $(n_1 - i - j, 1, 1)$ or $(n_1 - i - j, 0, 1)$ with $n_1 - i - j \geq 2$. Hence, by inspecting the first heap and reducing it to height 0 or 1, respectively, R can win. For $j = i - 1$, R ends up in $(1, 1, 1)$ or $(1, 0, 1)$, having value -1 or 1, respectively. Taking $j = i$ leads to $(0, 1, 1)$ or $(0, 0, 1)$, having values 1 and -1 .

If L chooses to inspect the second heap, she must empty it if it was of height 1, leading to $(n_1 - i, 0, 1)$ for R, who will win as $n_1 - i > 1$ by assumption. If it was already empty, L may choose another move on $(n_1 - i, 0, 1)$ and thus win. Emptying the third heap leads to either $(n_1 - i, 1, 0)$ or $(n_1 - i, 0, 0)$, which are both winning for R.

Hence, in summary, the value obtained by L playing any strategy against the fixed strategy of R is

$$p_{n_1} \left(-\frac{1}{2} + \frac{1}{2} \right) + p_{n_1-1} \left(\frac{1}{2} - \frac{1}{2} \right) \\ + \sum_{i=1}^{n_1-2} p_i \left(\frac{1}{2} \left(\sum_{j=1}^{n_1-i-2} -q_j^{n_1-i} - q_{n_1-i-1}^{n_1-i} + q_{n_1-i}^{n_1-i} - r_1^{n_1-i} - r_2^{n_1-i} \right) \right)$$

$$\begin{aligned}
& + \frac{1}{2} \left(\sum_{j=1}^{n_1-i-2} -q_j^{n_1-i} + q_{n_1-i-1}^{n_1-i} - q_{n_1-i}^{n_1-i} + r_1^{n_1-i} - r_2^{n_1-i} \right) \\
& = - \sum_{i=1}^{n_1-2} p_i \left(\sum_{j=1}^{n_1-i-2} q_j^{n_1-i} + r_2^{n_1-i} \right)
\end{aligned}$$

Maximizing her payoff, L will thus choose the variables such that the value becomes 0. This proves that $v(\text{FSN}(n_1, n_2, 1)) = 0$ and concludes the first part.

Next, let $n \in \mathbb{N}^d$, $d \geq 3$ be such that $n_1, n_2 \geq 2$ and $n_3 = \dots = n_d = 1$. We will again show that the value of this game is 0. We apply induction on d . The base case is $\text{FSN}(n_1, n_2, 1)$, which was shown to have value 0 above. Hence, let $d \geq 4$. If L picks any of the heaps except the first two, R finds himself in the situation $d - 1$ immediately and we are done. Remains to show that L cannot obtain a payoff larger than 0 by choosing any other strategy.

Denote by 1^k the vector $(1, 1, \dots, 1) \in \mathbb{N}^k$. Abusing notation, we may thus represent the starting configuration by $(n_1, n_2, 1^{d-2})$. Without loss of generality, suppose L picks i chips from the first heap in the first turn with probability p_i , $i \in \{1, \dots, n_1\}$. Again, we give an explicit strategy for R which prevents L from scoring higher than 0.

On his first move, R is given the information set $[\overline{n_1 - 1}, n_2, 1^{d-2}]$. With probability $\frac{1}{2}$, R reduces the second heap to height 1, and with equal probability, R empties the heap. L is thus faced with the information set $[n_1 - i, \overline{n_2 - 1}, 1^{d-2}]$.

For the case $n_1 - i = 1$, this gives either the state $(1, 1, 1^{d-2})$ or $(1, 0, 1^{d-2})$, which yields value $(-1)^{d-1}$ or $(-1)^d$, respectively. For $n_1 - i = 0$, it results in the state $(0, 1, 1^{d-2})$ or $(0, 0, 1^{d-2})$, giving value $(-1)^d$ or $(-1)^{d-1}$, respectively.

Now, suppose $n_1 - i > 1$, and let L remove i chips from the first heap with probability $q_j^{n_1-i}$, $j \in \{1, \dots, n_1 - i\}$. L inspects the second heap with probability $r_1^{n_1-i}$, and, without loss of generality, L empties the third heap with probability $r_2^{n_1-i}$. For $j \in \{1, \dots, n_1 - i - 2\}$, R is given the state $(n_1 - i - j, 1, 1^{d-2})$ or $(n_1 - i - j, 0, 1^{d-2})$, with $n_1 - i - j \geq 2$. Hence, by inspecting the first heap and reducing it to height $\mathbb{1}_{\{d \text{ even}\}}$ or $\mathbb{1}_{\{d \text{ odd}\}}$, respectively, R can win. For $j = i - 1$, R is led to $(1, 1, 1^{d-2})$ or $(1, 0, 1^{d-2})$, having value $(-1)^d$ or $(-1)^{d-1}$, respectively. Taking $j = i$ gives rise to $(0, 1, 1^{d-2})$ or $(0, 0, 1^{d-2})$, having values $(-1)^{d-1}$ and $(-1)^d$.

If L chooses to inspect the second heap, she must empty it if it contained a chip, leading to $(n_1 - i, 0, 1^{d-2})$ for R, who will win as $n_1 - i > 1$ by assumption. If it was empty, L must choose another move on $(n_1 - i, 0, 1^{d-2})$ and thus win. Emptying the third heap leads to either $(n_1 - i, 1, 0, 1^{d-3})$ or $(n_1 - i, 0, 0, 1^{d-3})$, which are both winning for R.

Hence, in summary, the value obtained by L playing any strategy against the fixed strategy of R is

$$\begin{aligned}
& p_{n_1} \left(\frac{1}{2} (-1)^d + \frac{1}{2} (-1)^{d-1} \right) + p_{n_1-1} \left(\frac{1}{2} (-1)^{d-1} + \frac{1}{2} (-1)^d \right) \\
& + \sum_{i=1}^{n_1-2} p_i \left(\frac{1}{2} \left(\sum_{j=1}^{n_1-i-2} -q_j^{n_1-i} + q_{n_1-i-1}^{n_1-i} (-1)^d + q_{n_1-i}^{n_1-i} (-1)^{d-1} - r_1^{n_1-i} - r_2^{n_1-i} \right) \right. \\
& \left. + \frac{1}{2} \left(\sum_{j=1}^{n_1-i-2} -q_j^{n_1-i} + q_{n_1-i-1}^{n_1-i} (-1)^{d-1} + q_{n_1-i}^{n_1-i} (-1)^d + r_1^{n_1-i} - r_2^{n_1-i} \right) \right)
\end{aligned}$$

$$= - \sum_{i=1}^{n_1-2} p_i \left(\sum_{j=1}^{n_1-i-2} q_j^{n_1-i} + r_2^{n_1-i} \right)$$

Again maximizing her payoff, L will make sure to obtain a value of 0. This concludes the second part of the proof.

For the final part, let $n \in \mathbb{N}^d$, $d \geq 3$ be such that $n_k > 1$ for more than one $k \in \{1, \dots, d\}$. Again, we show that the value is 0. Suppose without loss of generality that $n_1, \dots, n_c > 1$ and $n_{c+1} = \dots = n_d = 1$ for some $c \in \{2, \dots, d\}$. We use induction on c . The base case $c = 2$ is proven above. Hence, consider $c > 2$.

Suppose first that L plays on one of the first c heaps. As a fixed strategy for R in this case, we let him inspect this heap, and empty it if it was not yet so. This leads to a configuration with $c - 1$ either for L or R to move on, resulting in value 0 in any case.

Next, suppose that L plays on one of the last $d - c$ heaps. R can then force a value of 0 by emptying the first heap and moving to a situation with $c - 1$ in this case. \square

Note that a very similar result holds for the misère version of the game. Denoting $\text{FSN}^-(n)$ to be the misère version of Fuzzy Schrödinger Nim, we obtain the following.

Theorem 4.2. *Let $n \in \mathbb{N}^d$. Then*

$$v(\text{FSN}^-(n)) = \begin{cases} -1 & \text{if } \max_{k=1, \dots, d} n_k = 1 \text{ and } d \text{ is odd,} \\ 1 & \text{if } \max_{k=1, \dots, d} n_k = 1 \text{ and } d \text{ is even,} \\ & \text{or if } n_k > 1 \text{ for precisely one } k, \\ 0 & \text{otherwise.} \end{cases}$$

Indeed, in the proof above, the signs of the base cases are flipped. For the explicit strategies described for R in the proof, we can mirror the behaviour for d being odd or even, resulting in the same value of the game as before, being zero. For this misère variant of the game, we find that the values somewhat resemble those for the misère version of the regular game of Nim.

5. KRIEGSPIEL NIM

Finally, we consider the Nim variant inspired by Kriegspiel, described in Ciancarini and Favini (2010). Recall that, for $n \in \mathbb{N}^d$, we denote this variant of the game with starting position n by $\text{KN}(n)$. We only consider the game with at most two heaps.

For $d = 1$, the game is trivially won by Left, removing all chips on her first turn. For $\text{KN}(n, 1)$, Left removes all but one chip from the first heap on her first turn, winning the game. For the other cases with $d = 2$, we have the following result.

Theorem 5.1. *Let $n_1, n_2 \in \mathbb{N}$, $n_1, n_2 \geq 2$. Then $v(\text{KN}(n_1, n_2)) = 0$.*

Proof. We give an explicit Nash equilibrium pair, depicted in Fig. 5. On the first turn, L reduces the first heap to size 1 or 0, both with probability $\frac{1}{2}$. Next, R tries to empty either heap with probability $\frac{1}{2}$. The attempt to empty the first heap will fail, after which R reduces the second heap to size 1.

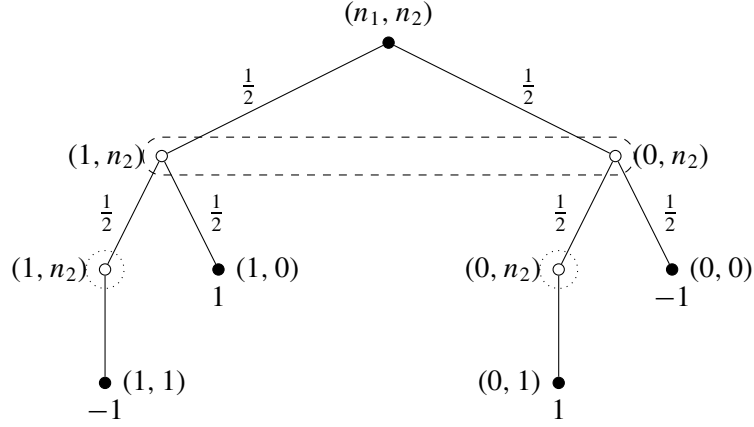


Fig. 5. A Nash equilibrium for $KN(n_1, n_2)$.

Suppose R deviates. Because we assume the strategy of L to be fixed, we may assume R to know that he is either in the state $(1, n_2)$ or $(0, n_2)$, both with probability $\frac{1}{2}$. Hence, picking any move which tries to remove more than 1 chip from the first heap needs not be considered. Now, suppose R tries to pick a single chip from the first heap with probability p , and picks k chips from the second heap with probability $q_k, k = 1, \dots, n_2$.

Moving from $(0, n_2)$, trying to move on the first heap will fail, telling R that the heap is empty. Hence, R may proceed to empty the second heap and win. Also, if R removes all n_2 chips from the second heap immediately, he wins. Otherwise, he leaves only chips on the second heap for L, who knows that the first heap is empty. Therefore, L can and will empty the second heap on her next turn and therewith win.

Now, consider the case where R moves from $(1, n_2)$. We define the strategy of L for her next turn(s) as follows: first, L will try to remove n_2 chips from the second heap. If this fails, she will try to remove 1 chip from the second heap. Finally, if this fails, she removes the single chip from the first heap. If and only if R has removed the single chip from the first heap, L will be able to remove n_2 chips from and therewith empty the second heap, winning the game. If R emptied the second heap, L will empty the first, again winning. If R removed all but one chip from the second heap, L is faced with $(1, 1)$ losing in any case. Otherwise, L moves from $(1, n_2 - k)$ to $(1, n_2 - k - 1)$. For $k = n_2 - 2$, this leads R to the state $(1, 1)$, where he loses. For any other $k \in \{1, \dots, n_2 - 3\}$, R can proceed by reducing the second heap to size 1, giving the state $(1, 1)$ to L, and winning.

Hence, the payoff of the game now becomes

$$\frac{1}{2} \left(p - \sum_{k=1}^{n_2-3} q_k + q_{n_2-2} - q_{n_2-1} + q_{n_2} \right) + \frac{1}{2} \left(-p + \sum_{k=1}^{n_2-1} q_k - q_{n_2} \right) = q_{n_2-2},$$

so, minimizing, R will choose $q_{n_2-2} = 0$, yielding a value of 0. Hence, deviating does not lead to a better payoff for R.

Next, suppose L deviates. Let L remove k chips from the first heap with probability $p_k, k = 1, \dots, n_1$, or k chips from the second heap with probability $q_k, k = 1, \dots, n_2$. From $(n_1 - k, n_2)$, R will play to either $(n_1 - k, 1)$ or $(n_1 - k, 0)$, both with probability $\frac{1}{2}$. From $(n_1, n_2 - k)$, R will move to $(0, n_2 - k)$ or $(1, n_2 - k)$, again both with probability $\frac{1}{2}$. We will analyse the strategies going from $(n_1 - k, n_2)$; the strategies moving from $(n_1, n_2 - k)$ are symmetrical.

Similar to the discussion above, L knows that she is in either $(n_1 - k, 1)$ or $(n_1 - k, 0)$, both with probability $\frac{1}{2}$. Therefore, we may discard all moves trying to take more than 1 chip from the second heap. Now, let L remove i chips from the first heap with probability $r_i^{n_1-k}$, $i = 1, \dots, n_1 - k$, and let L try to remove the single chip in the second heap with probability s . If this fails, she proceeds to empty the first heap and win.

For R, we define the follow-up strategies as follows. If he emptied the second heap on his first move, he will empty the first heap and win the game on his next turn. If he left a chip in the second heap, he will try to remove it. If this fails, he empties the first heap and wins. If it succeeds, L can proceed by emptying the first heap and winning.

All in all, this leads to the following total payoff, considering only the strategies in which L moves on the first heap in the first turn:

$$\sum_{k=1}^{n_1-2} p_k \left(\frac{1}{2} \left(\sum_{i=1}^{n_1-k-1} r_i^{n_1-k} - r_{n_1-k}^{n_1-k} - s \right) + \frac{1}{2} \left(- \sum_{i=1}^{n_1-k-1} r_i^{n_1-k} + r_{n_1-k}^{n_1-k} + s \right) \right) = 0.$$

Note that we have excluded the cases $k = n_1 - 1$ and $k = n_1$, which both clearly lead to value 0, as well. Moreover, moving on the second heap in any way leads to the same payoff. Hence, deviating gives no larger payoff for L and we are done. \square

To conclude, we consider one more variant of Kriegspiel Nim. In this variant, during a player's turn, a move still consists of trying to remove i chips from the j -th heap. If this is possible, it is done; otherwise, as many chips as possible are removed and the heap is left empty. In any case, the turn is passed, the other player not being informed of anything. It can be proven, analogous to the arguments above, that for this variant, the values for up to two heaps coincide with the values of Kriegspiel Nim.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we have considered three non-perfect information variants of the game Nim. For the first variant, Schrödinger Nim, we provided a partial characterization, and a set of structural results. For the second variant, Fuzzy Schrödinger Nim, we provided a complete characterization. For the third variant, Kriegspiel Nim, we gave some preliminary results.

A full solution to the first variant remains elusive, and is a natural direction for further research. One might try to replicate Lemma 3.2 with tighter bounds for higher heaps, for example. Moreover, it would be interesting to look at the *misère* version of this first variant.

The aim of introducing these variants is to generalize the theory of combinatorial games to the class of non-perfect information games. With this aim in mind, the third variant of Kriegspiel Nim is the most widely applicable. Therefore, not only is it interesting to further research this game, but using a similar setup, one might create non-perfect information variants of all combinatorial games. Consider, e.g., the game of Hackenbush. A move will now consist of asking the umpire whether it is possible to remove an edge. If so, it is done; otherwise, the player has to try and remove a different edge. Similarly, in Domineering, a move might be to try to place a stone until the placement is successful. By analysing these non-perfect information variants of different combinatorial games and comparing the results, we hope to distinguish some structure akin to that in the class of combinatorial games.

REFERENCES

- Albert, M.H., Nowakowski, R.J. & Wolfe, D. (2019). *Lessons in Play* (2nd ed.). CRC Press.
- Berlekamp, E.R., Conway, J.H. & Guy, R.K. (1982). *Winning Ways for Your Mathematical Plays* (2nd ed. 2001–2004). Academic Press.
- Bowling, M., Burch, N., Johanson, M. & Tammelin, O. (2017). Heads-up limit hold'em poker is solved. *Communications of the ACM*, 60(11), 81–88. doi:[10.1145/3131284](https://doi.org/10.1145/3131284).
- Ciancarini, P. & Favini, G.P. (2007). Representing Kriegspiel states with metapositions. *IJCAI*, 07, 2450–2455.
- Ciancarini, P. & Favini, G.P. (2010). Monte Carlo tree search in Kriegspiel. *Artificial Intelligence*, 174, 670–684. doi:[10.1016/j.artint.2010.04.017](https://doi.org/10.1016/j.artint.2010.04.017).
- Cincotti, A. & Iida, H. (2007). *The Game of Synchronized Cutcake*. 2007 IEEE Symposium on Computational Intelligence and Games.
- Ferguson, T.S. (2014). *Game Theory*. Los Angeles: Mathematics Department, University of California.
- Hayward, R.B. & Toft, B. (2019). *Hex: The Full Story*. CRC Press.
- Huggan, M. & Nowakowski, R.J. (2018). Simultaneous combinatorial game theory. Arxiv preprint.
- Kuhn, H.W. (1953). Extensive games and the problem of information. In *Contributions to the Theory of Games II*. Annals of Mathematics Studies (Vol. 28, pp. 193–216). Princeton University Press.
- Moravčík, M., Schmid, M., Burch, N., et al. (2017). DeepStack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337), 508–513. doi:[10.1126/science.aam6960](https://doi.org/10.1126/science.aam6960).
- Nisan, N., Roughgarden, T., Tardos, É. & Vazirani, V.V. (2007). *Algorithmic Game Theory*. New York, New York: Cambridge University Press.
- Siegel, A.N. (2013). *Combinatorial Game Theory*. American Mathematical Society.
- Silver, D., Schrittwieser, J., Simonyan, K., et al. (2017). Mastering the game of Go without human knowledge. *Nature*, 550, 354–359. doi:[10.1038/nature24270](https://doi.org/10.1038/nature24270).
- Uiterwijk, J.W.H.M. (2016). 11 × 11 domineering is solved: The first player wins. In *Computers and Games*. CG 2016. Lecture Notes in Computer Science (Vol. 10068, pp. 129–136). Springer.
- Von Stengel, B. (2002). Computing equilibria for two-person games. In *Handbook of Game Theory with Economic Applications* (Vol. 3, pp. 1723–1759). Amsterdam: Elsevier.