The American Mathematical Monthly

ISSN: 0002-9890 (Print) 1930-0972 (Online) Journal homepage: www.tandfonline.com/journals/uamm20

A Non-Terminating Game of Beggar-My-Neighbor

Brayden Casella, Philip M. Anderson, Michael Kleber, Richard P. Mann, Reed Nessler, William Rucklidge, Samuel G. Williams & Nicolas Wu

To cite this article: Brayden Casella, Philip M. Anderson, Michael Kleber, Richard P. Mann, Reed Nessler, William Rucklidge, Samuel G. Williams & Nicolas Wu (2025) A Non-Terminating Game of Beggar-My-Neighbor, The American Mathematical Monthly, 132:10, 978-994, DOI: 10.1080/00029890.2025.2558484

To link to this article: https://doi.org/10.1080/00029890.2025.2558484

© 2025 The Author(s). Published with license by Taylor & Francis Group, LLC.

Published online: 20 Oct 2025.

Submit your article to this journal

Article views: 726

View related articles

View Crossmark data

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=uamm20

A Non-Terminating Game of Beggar-My-Neighbor

Brayden Casella, Philip M. Anderson, Michael Kleber, Richard P. Mann, Reed Nessler, William Rucklidge, Samuel G. Williams, and Nicolas Wu

OPEN ACCESS

Abstract. We demonstrate the existence of a non-terminating game of Beggar-My-Neighbor discovered by lead author Brayden Casella. We detail the method for constructing this game and identify a cyclical structure of 62 tricks that is reached by 30 distinct starting hands. We further present a short history of the search for this solution since the problem was posed, and a record of previously found longest terminating games. The existence of this non-terminating game provides a solution to a long-standing question which John H. Conway called an "anti-Hilbert problem."

1. INTRODUCTION. Beggar-My-Neighbor (BMN)[1] is a simple deterministic card game for two players. The rules, as described in the 1911 edition of *Encyclopaedia Britannica* are as follows:

'An ordinary pack is divided equally between two players, and the cards are held with the backs upwards. The first player lays down his top card face up, and the opponent plays his top card on it, and this goes on alternately as long as no court-card appears; but if either player turns up a court-card, his opponent has to play four ordinary cards to an ace, three to a king, two to a queen, one to a knave[2]*, and when he has done so the other player takes all the cards on the table and places them under his pack;*[3] *if, however, in the course of this playing to a court-card, another court-card turns up, the adversary has in turn to play to this, and as long as neither has played a full number of ordinary cards to any court-card the trick continues. The player who gets all the cards into his hand is the winner.'*

Anyone playing BMN repeatedly will notice that some games are much longer than others. It is a natural question to ask whether there is a game which can continue indefinitely, and has surely occurred independently many times, but notably by John H. Conway during his time as a student at Cambridge [**1**]. Since there are only finitely many ways to arrange a deck of cards, a game continues forever if and only if at some point the game returns to exactly a situation it has been in before; i.e., the game play enters a loop. Conway further wrote, "The question of whether it always terminates is one of my 'anti-Hilbert' problems. [Hilbert's were some problems for mathematicians

doi.org/10.1080/00029890.2025.2558484
MSC: 00A08
© 2025 The Author(s). Published with license by Taylor & Francis Group, LLC.
This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

[1]Additional names for this game include Beggar *Your* Neighbor, Strip Jack Naked, Drive the Old Woman to Bed, and many others.

[2]In this paper we use the more typical terminology for the court (or face) cards: Ace (A); King (K); Queen (Q) and Jack (J).

[3]The player who takes a trick is also the one who lays down the first face-up card to begin the subsequent trick.

to work on during this century. Mine are problems we should NOT be working on during the next!]"[4] [1, 3–5].

Despite this inauspicious framing, a number of individuals (including all the authors of this paper) have since searched for a non-terminating game, and in the process have discovered games of increasing length. In this paper we give a short historical perspective on the search and demonstrate the existence of a non-terminating game.

History of the search. The computer-assisted hunt for a non-terminating game of BMN first appears in print in [6]. Beasley reports on the statistical results of ten thousand randomly-dealt games, and notably observes that the game's length seems to have a half-life: "if a game is down to two players but has not yet finished, the probability is about even that it will still not have finished within a further twenty tricks." He further notes that if the game were instead a *random* map on possible game states with a 20-trick half-life, then finding a loop in the game state would be like looking for a needle in a haystack: a crude probabilistic argument says that there is over a 90% chance of a loop existing, but the chance that a randomly-chosen starting position leads into a loop is around $1/10^{22}$.

In 1991, inspired by (or perhaps "not deterred by") this needle-in-a-haystack assessment, Rutgers University student Chris Long posted to the usenet newsgroup rec.puzzles, kicking off what he called *The Great Usenet BYN Contest*[7]. Long inaugurated the contest with the announcement that after a computer search of around 50 million random games, he had not found any that ran forever, but had found a game which ended after 538 tricks, and he challenged rec.puzzles contributors to find an infinite game or, failing that, to beat his record. Long also reported that "Jim Propp has informed me that John Conway has offered $100 to anyone who can exhibit an initial BYN hand that has an infinite cycle." The contest ran for six weeks, and was won by Mark S. Manasse of Digital Equipment Corporation, who found a 713-trick game [8], probably having examined on the order of 100 billion random games.

This unsolved problem gained greater prominence in 1999 with the publication by Marc Paulhus of an investigation into the structure of BMN, including the construction of non-terminating games in reduced sets of cards [1], and the presentation of early records for the longest known terminating games of BMN with a full deck. Published by *The American Mathematical Monthly* as part of its Unsolved Problems section, this brought the problem and its origins into wider visibility in the math community.

An exhaustive test of all possible deals is computationally implausible: there are $\binom{52}{4}\binom{48}{4}\binom{44}{4}\binom{40}{4} \simeq 6.54 \times 10^{20}$ possible distinct starting configurations. Note that this number ignores the details of a deck of cards that are irrelevant to the game; only the positions of the face cards and aces affect the play, the positions of spot cards and the suits of the face cards and aces are completely interchangeable. As will be seen below, even very substantial computing resources can only hope to explore a small part of this space. Therefore three approaches have suggested themselves: (i) statistical optimization; (ii) brute force random search; and (iii) direct construction of a non-terminating game.

It is straightforward to computationally simulate many randomly-dealt games of BMN and to record the number of cards and tricks played during these games. Figure 1 shows the frequency of different game lengths in a sample of 10^7 randomly-dealt games. This closely resembles an exponential distribution, except in the case of very short games. The apparently-exponential distribution of game lengths suggests that although the progress of the game is deterministic, from a coarse-grained perspective

[4] We know of no definitive list of Conway's anti-Hilbert problems, but his Thrackle Conjecture is a second example; see [2].

(one in which we do not see the details of the packs) any ongoing game has a constant probability of "collapsing" as it progresses, and thus the fact that a game has endured for a certain amount time has no predictive power over its future length. The tail of this distribution closely matches Beasley's earlier observation of a half life of ∼20 tricks.

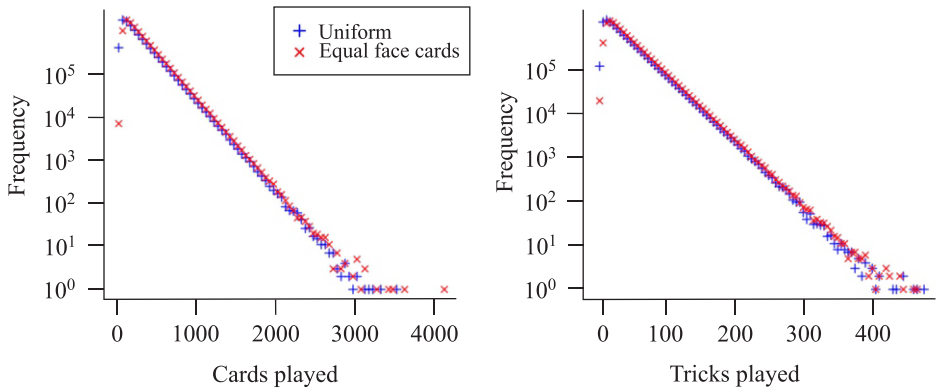

Figure 1. Frequency distribution of game lengths from 10^7 randomly-dealt games, either with uniformly random deals (blue + points) or restricted to deals in which both players have an equal share of the face cards (red × points). Apart from very short games, the frequencies in all cases closely resembles an exponential distribution (note the log-scale on the y-axis). Although dealing an equal share of face cards marginally increases expected game length, it does not change the decay rate of the exponential tail of the distribution.

A statistical approach seeks to identify coarse-grained characteristics of starting packs that are associated with greater game length. An intuitively-sensible strategy is to ensure that each player is dealt the same number of each face card, such that the game is more evenly-balanced. As shown in Figure 1 this strategy does have the effect of increasing the mean length of randomly-generated games. However, this is mostly achieved by selectively avoiding the very shortest games, and the exponential tail of the distribution appears to decay at the same rate as when games are dealt uniformly at random from all possible starting packs. As such this strategy can only very marginally increase the expected size of the longest game found over n randomly-generated examples, and in our experience is an ineffective method for discovering longer games in the far tails of the distribution. Similar statistical approaches such as identifying specific individual card placements associated with higher mean game lengths also proved unsuccessful at systematically generating games of greater extreme length.

A large part of the search was subsequently carried out by brute force: simulating many randomly-dealt games in the hope of finding one that did not terminate, or as a lesser goal identifying one longer than any yet discovered. A significant innovation was to take found long games and play them "backwards"; that is, to identify earlier card configurations that result in the same configuration as those starting games do, such that this could represent an alternative starting position that produces a longer game. Although playing a game forwards is an entirely deterministic process, playing backwards involves some nondeterminism; families of different starting hands are able to converge on a particular state. These cases arise when looking through the history of a player's hand. When there is a matching amount of number cards after a face card, this could be because they won a turn, but it might also be due to some residual play. An example of games which lead to the same state can be seen in Appendix B. Hence one must explore and keep track of an expanding tree of potential former states until none of these can be played backwards any further.

Nonetheless, despite this innovation, the search remained primarily constrained by the available computing power and time. To illustrate the scale of this search, Marc

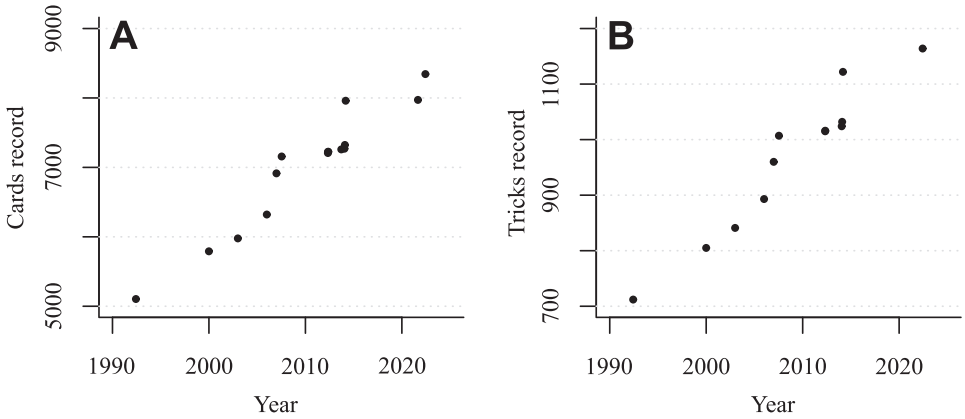

Figure 2. Historical records since 1992 for the longest known terminating game of BMN in terms of cards played (A) and tricks (B). A full list of historical records known to the authors is presented in Appendix A.

Paulhus played 3.2×10^9 games prior publishing his initial findings [**1**]; Richard Mann and Nicolas Wu (Appendix A) played c. 10^{13} games to find a record in 2007, running as a background process on between 5 and 10 desktop workstations in the Department of Engineering Science at the University of Oxford over a period of several weeks; William Rucklidge (Appendix A) played 3.3×10^{15} games in 2014 using 1000–8000 idle cores in Google's fleet, at a rate of about 10^9 games per hour per core; Reed Nessler played 3.5×10^{15} games over several years on a disused workstation capable of 7×10^{10} games per hour.

Figure 2 shows the historical progress of the records for known longest games, in terms of either cards played (A) or tricks (B). We stress that these are records known to us, which have been recorded via personal correspondence. Despite some stops and starts, progress in this record was broadly linear in time over the long term. This is straightforwardly explicable in terms of a simple model of random search. First, let us assume that game lengths are exponentially distributed with mean μ, as indicated in Figure 1 (this can apply to either cards or tricks). Assuming that a large number (n) of random games have been played, the expected length of the longest game (r) is given via standard order statistics as:

$$\langle r \rangle = \mu \sum_{i=1}^{n} \frac{1}{n-i+1} = \mu H_n \simeq \mu(\gamma + \ln(n)), \tag{1}$$

where H_n is the nth harmonic number and γ is the Euler-Mascheroni constant. If we further assume that computing power increases exponentially over time and making the highly-simplifying assumption that attention to solving the problem remains constant (i.e., a constant fraction of available compute is dedicated to the task), then:

$$n = Ae^{kt}, \tag{2}$$

for some constants A and k. Substituting this value of n into the earlier expression gives:

$$\langle r \rangle \simeq \mu(\gamma + \ln A + kt), \tag{3}$$

showing that the longest game is expected to grow linearly with time t.

2. CONSTRUCTING A NON-TERMINATING GAME. Playing games with a standard deck of cards can be time consuming and provides little direct insight for

understanding the game. When the goal is to find a non-terminating game, it is also unclear how looking at terminating games will give insight into non-terminating games. Therefore a good starting point is to find non-terminating games with the fewest number of cards.

Many smaller non-terminating decks with different distributions are easy to find. Consider two starting hands of a 6-card game: `J--` and `-J-` where the first hand begins play (hereafter notated as `1. J-- 2. -J-`, where any number card is notated as "-"). The starting trick will transfer `J-` to the first hand, then infinitely add `-J-` to alternating hands. We can quickly find more complex solutions through exhaustive search of smaller decks.

Consider this 10 card non-terminating game:

```
1. --Q-J-
2. Q-J-
```

Continuing play results in this state being reached again, resulting in an infinite loop. This cycle is shown here (note, for each configuration, the number in parentheses is the player who starts play, i.e., the winner of the previous trick):

```
1. --Q-J- 2. Q-J-     (1)
1. J-     2. --Q-Q-J- (2)
1.--J-    2. Q-Q-J-   (1)
1. --Q-J- 2. Q-J-     (1)
```

Other examples of non-terminating games in reduced decks are given by Paulhus [**1**]. Though solutions have been found using random search for decks up to 40 cards [**9**] with various numbers of face cards, solve time grows rapidly, and brute force becomes impractical.

Expansion, mutation, and the first non-terminating but unbalanced game. Due to the repeating sequence of tricks in the `1. J-- 2. -J-` game, it is possible to expand this simple deck to a slightly larger non-terminating game. Because the added tricks repeat, attaching those tricks to the end of the initial hands preserves the repeated sequence, and keeps the game non-terminating. In this case, it is done by doubling each player's hand which makes the new game `1. J--J-- 2. -J--J-`. More complicated games will have a longer repeating sequence of added tricks and a different way to expand into larger games. More generally, playing any non-terminating game can be thought of as a list of states, with each state consisting of both player's hands:

$$I_1, I_2 \ldots, I_m, L_1, L_2, \ldots, L_n, L_1, L_2 \ldots L_n, \ldots.$$

Here, each successive state corresponds to one after a trick is added to a hand. Each state I_i where $i \in 1 \ldots m$ corresponds to an initial state before the loop, and each state L_j where $j \in 1 \ldots n$ corresponds to a state that is in the loop, which has a periodicity of n states. Each state eventually leads to a non-terminating game, but only each L_j will be repeated infinitely, where the repeated states are:

$$L_1, L_2, \ldots, L_n.$$

All tricks added during a single state loop can be concatenated to the end of the respective hands at state L_1. This new game will also be non-terminating. An additional step can be taken to expand further, where this expanded game can be attached any amount of times to make another non-terminating game for each addition. The decks

created from this method are often not balanced (games where the length of the starting hands of each player are equal), but cutting the hands at equal length does often yield a non-terminating game.

This way, finding small non-terminating games with at least one of each face card might be able to be expanded into a standard deck which is non-terminating. To generate smaller games, an exhaustive search of balanced games up to 26 cards and various amounts of each face card can be generated and non-terminating games saved. A half-sized deck with half the face cards might be simply expanded into a standard deck, but this is not the case as there is no non-terminating game with this distribution [1].

Forcing the games to be balanced is a severe limitation. To move into the unbalanced space, the thousands of saved non-terminating games can be expanded using the method described above. Each possible combination within those expansions of unbalanced games also tested for termination. Filtering the results for non-terminating decks which contain 4 of each face card eventually yields this deck:

```
1. --Q-K-J-----------------KJ--Q----A-Q-A-A---J--Q-K-A-K
2. J-
```

There are 39 instead of 36 number cards but the three number cards in the `A---J` sequence can be removed to make a non-terminating standard deck. This game does not become balanced at any state in the loop. Moreover, playing backwards to explore all possible earlier states that lead to the loop yields no balanced deck. A different deck is needed to construct a cycle that is accessible from an initial deal.

Small changes to non-terminating games can often be found which create new non-terminating games. Three operations can be used to do this. For the first, at every point in the deck, each type of card (number card, jack, queen, king, and ace) is temporarily placed into the deck at that spot, then the game is played and tested for termination. If the game is non-terminating, that new deck is saved. The second operation is removal of cards, where each card in the deck is temporarily removed, then the deck tested and saved if non-terminating. The final operation is swapping one card for another. Similarly to the previous two operations, each card in the deck is temporarily changed to every other card and each change tested for termination. All points to split the deck between player 1 and player 2 are also tested for each operation. The speed of these operations is an issue so only a maximum of three additions and removals at a time were performed. Through the operations and manual changes, this non-terminating standard deck can be found:

```
1. --------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A-A-AJ-
2. J-
```

At no point in the state loop of this game do the hands become balanced, nor do any backtracked predecessors of the loop.

It is very difficult to create further non-terminating standard games using the operations and manual changes from the two games found. A tree search through the operations also failed to yield any interesting non-terminating games.

Deck structure and Casella's Ansatz. The two non-terminating standard games have a more general set of repeating states. Consider one of the games shown here:

```
1. --------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A-A-AJ-
2. J-
```

Selecting specific states (in order, each 4 tricks apart) from the full trace of this game, we observe:

```
1. --------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A-A-AJ- 2. J-

1. --Q-Q-Q-Q-K-KJ--K-K-A-A-A-AJ--------------------J- 2. J-

1. --K-K-A-A-A-AJ--------------------J--Q-Q-K-Q-K-QJ- 2. J-

1. --------------------J--Q-Q-K-Q-K-QJ--A-K-A-K-A-AJ- 2. J-
```

Notice how player 1's hand mostly looks as though the deck is only being "slid" over until it reaches a jack. Sliding this way three times will result in an almost identical deck to the original, where all that is different is the order of a few face cards. Continuing play will return the face cards to the original order. The full trace of the solution can be found in Appendix D, where this can also be seen. Some different sequences will not be added to the end of player 1's hand as it was originally, but instead have their own loop and will return to the original sequence when the game continues. Importantly, player 2's hand will always return to J− before moving on to play through the next sequence in player 1's hand. These jack-terminated sequences, or "pieces" as they will be referred to, are therefore independent from one another and can be interchanged in any order to create a non-terminating deck. Due to this, player 2's hand can be seen only as a place to temporarily hold part of the current piece being played.

Inspired by this observation, we consider this promising-seeming shape for more non-terminating decks:

Casella's Ansatz. To search for an infinite game of BMN:

1. Look for games which pass through the state

   1. Δ_1 J Δ_2 J Δ_3 J−
   2. J−

   where deck pieces Δ_1, Δ_2, and Δ_3 behave independently, and game play proceeds by cycling these three deck pieces after t_1, t_2, and t_3 tricks, respectively. Some small internal permutations of face cards within each Δ_i may occur without disrupting the overall pattern.
2. To find a balanced starting position with 26 cards in player 2's hand, look for:
   (a) A deck piece Δ_i which can be formed by forward play from a state with more cards in player 2's hand (which we may search for by playing pieces backwards), and
   (b) A deck piece Δ_j which moves a lot of cards over to player 2's hand in the middle of its forward play (so that the balanced initial state will represent the middle of playing through Δ_j).

The precise non-terminating games found by the search up to this point are no longer relevant except as inspiration for this new method. Finding the correct combi-

nation of pieces is the primary focus and what leads to the balanced non-terminating standard deck.

In these pieces, the most important aspect is the amount of face cards. Since number cards can often be added and removed from the same piece, and face cards can often be swapped for any other face card, the amount of strictly unique pieces can become overwhelming. The `--Q-Q-Q-Q-K-K` piece in particular can lead to many strictly unique pieces due to this. For example:

`--Q-Q-Q-Q-K-K`,
`--Q-Q-Q-Q-K-A`,
`--Q-Q-Q-Q-K-A-`, and
`--Q-Q-Q-Q-KK`

are all different pieces, but have the same amount of face cards with only minor mutations. The number of different pieces generated through face card swapping is 3^6 from the `--Q-Q-Q-Q-K-K` piece. With a number card added there are another 3^6 unique pieces added with face card swapping. Since there are many ways to add and remove number cards from this piece, there are thousands of pieces which are only slightly different. Saving all of these for manual construction would not be useful. Instead, a single one of these pieces can be saved and the operations used to find the variations. Due to this, pieces which only differ through the operations (or repeated steps through the operations) are considered the same piece. For example, `--Q-Q-Q-Q-K-K` and `--Q-K-A-KQ-K-` are seen as the same piece.

The realization of the pieces also results in a natural template, or "skeleton," deck for checking if a sequence is a piece. Since all pieces have a structure which creates independence between the pieces in a game, any sequence which doesn't have this structure will likely not create a non-terminating game (and therefore not be a piece). The structure of the template originally used to test for pieces is:

1. `--J` TEST `J-`
2. `J-`

A problem arises when using this template. Many sequences lead to non-terminating games, but when they are put together, the resulting game is not non-terminating. This is due to the tested sequences leading to the template being non-terminating but not having the properties of a piece. Given that pieces need to be independent from one another, a simple way to prevent these false-positives is to add a known piece into the template. Doing this forces an interaction between the tested sequences and a known piece. If the template with this added piece is non-terminating, it is probable that the tested sequence has the desired properties. Given that the added piece 'filters' out the false-positives, the added piece will be referred to as a filter. The structure of the template is:

1. `--J` FILTER `J` TEST `J-`
2. `J-`

Not all pieces are suitable as the filter. Certain patterns in a piece can allow for specific groups of false positives. The piece `--K---A----AA` is a suitable filter. The template is now:

1. `--J--K---A----AAJ` TEST `J-`

```
2. J-
```

Any sequence can replace TEST in this template, just as `--K---A----AA` replaced FILTER. If the game is non-terminating with this sequence, then the sequence is a piece and can be saved. The template arises from the structure of the pieces, and any amount of pieces (each terminated by a jack) can be added to the start of player 1's hand to create a non-terminating game.

This template based on jacks can be substituted for one based on the other face cards. However, this requires the inclusion of jacks in the inserted pieces and when this is done, one finds that many number cards are required in those pieces, which forces any constructed deck using these pieces to have an excess. As such, the jack template is used here.

Testing all combinations of a standard deck is an exhaustive search of the entire space, which is too large for an exhaustive search, but using templates this search space is vastly reduced. The search space is not only reduced by the cards inherent to the template (four jacks and two number cards), but also due to a piece reducing the number of cards needed for the other pieces used in a deck. Therefore all combinations of increasing size are tested with the template. To quickly find small pieces, number cards are considered 0, queens 1, kings 2, and aces 3. Finding all possible combinations is the same as counting in base 4 and padding with a sufficient amount of zeros (a total of 40 digits were used). While testing these, the leftmost digit can be repeatedly taken off the sequence and the reduced sequence tested again until a face card is reached, then move onto the next 40 card sequence. While this works exactly as desired early on, once more than 4 of a single face card or more than 12 total face cards start being generated, those sequences being tested are invalid for a standard deck of cards. As more digits are being populated by face cards, the time it takes to get to the next digit will also take exponentially more time. While that stays true, multiset permutations can be used to only test sequences with a specific amount of each card. Another speedup can be made by not using aces in the sequences being tested. This is a viable method since the face cards in larger pieces can often be swapped for other face cards. For example, instead of testing sequences that include 2 aces, 4 kings, and 4 queens, testing sequences with 6 kings and 4 queens works very well in reducing the time it takes to find larger pieces. Then when a piece is found, checking replacement of the extra kings into aces can be done.

Ten different non-terminating standard decks, each part of different state loops, can be quickly constructed with the new pieces and an eventual 15 being created before the final game. Attempting backwards play at each state in the loop for all of these yields unbalanced games. All pieces and non-terminating standard games constructed from them can be found in Appendix C.

Playing backwards to find balance. Since the main focus is creating a balanced game, pieces need to be used which add as many cards to player 2's hand as possible. This can only be done through normal and backwards play of each piece. Beneficial backwards play for each piece mainly relies on subsequences within the piece. Looking back at the `--K---A----AA` piece used as the filter in the template, completing this with jack termination creates the non-terminating game:

```
1. --K---A----AAJ-
2. J-
```

To play backwards, what do the starting hands need to be to add an equivalent

subsequence of cards to the end of player 1's hand? Looking from right to left of player 1's hand, `-` is not a valid trick (because play would continue normally rather than the card added to a player's hand), so the first valid subsequence is `J-`. To add `J-` as a trick through normal play, the game would be `1. J 2. -`. These cards can be removed from the end of player 1's hand and the game which plays `J-` added to the start of the original hands. This creates:

```
1. J--K---A----AA
2. -J-
```

Playing this game forward one trick will create the original deck. If the same backwards play is attempted for player 1's hand now, the effort is immediately stopped since no valid trick ends with `A` in normal play. Therefore enough number cards are needed to be left for the next face card before the subsequence being removed. Going back to the original deck, the subsequence which does this is `AAJ-`. Playing this backwards creates:

```
1. AJ--K---A----
2. A-J-
```

Notice how four cards were removed from the end of player 1's hand, then two cards added to the start of each hand. The beneficial structure of the selected piece shows itself once backwards play is continued from here. The subsequence `A----` is selected to be removed from player 1's hand and played backwards, creating:

```
1. AAJ--K---
2. ----A-J-
```

Five cards are removed while only adding one to the start of player 1's hand, and four to player 2's hand. When the goal is to make player 2's hand larger, this is an extremely efficient subsequence. If the `K---` subsequence is selected for backwards play next, another four cards are removed from player 1's hand while only adding one card to the start of their hand. Not every piece moves so many cards into player 2's hand, and some pieces have subsequences which swap whose turn it is to play after playing backwards. This is disastrous if it is not the first subsequence in the piece and results in the entire piece being unusable to create a balanced deck (for all pieces found). As an example, this is a promising looking piece where the necessary `-KK--Q-K--Q--` subsequence in the piece swaps whose turn it is after backwards play:

```
--------K----KK--Q-K--Q--QQ
```

Now, all that's needed is a piece which allows for a balanced game. Eventually, this piece is discovered:

```
--------K---------Q-Q-K---Q-KKQ
```

Testing the piece shows that two of the face cards can be replaced with any other face card. Manually selecting the pieces `--K---A----AA` and `--`, then adding all three together with jack termination creates this deck:

```
1. --K---A----AAJ--J--------K---------Q-Q-K---Q-KAQJ-
2. J-
```

Playing forward naturally through the first piece, then backwards for the final piece yields the balanced non-terminating standard game:

```
1. ---K---Q-KQAJ-----AAJ--J--
2. ----------Q----KQ-J-----KA
```

The first non-terminating game and its family The non-terminating game constructed by the method above is given by the starting hands below, with player 1 being the starting player, and with the top of the hand being to the left.

```
1. ---K---Q-KQAJ-----AAJ--J--
2. ----------Q----KQ-J-----KA
```

This starting deal leads after 4 tricks to a cycle of 62 tricks of which the first repeating state is:

```
1. ---AAJ--J--------K---------Q-Q-K---Q-KAQJ-
2. -----KA-J-
```

with player 2 to play next. The full progress of the cycle is given in Appendix D.

By playing backwards from each state within the cycle, it is possible to identify a family of 30 possible starting deals, each of which leads to a point on this cycle and thus each of which represents a non-terminating game. These are illustrated in Figure 3.

3. CONCLUSION. We have demonstrated the existence of a non-terminating game of Beggar-My-Neighbor, providing a solution to the "anti-Hilbert problem" posed by John H. Conway. Having identified a cyclical game, we were able to further construct a family of related solutions that converge to the same cycle. Although the original open problem is now resolved, several questions remain open. First, how many different cycles exist, and which of these can be reached from a standard deal of 26 cards for each player? Second, how many balanced non-terminating games exist? Third, what is the length of the longest terminating game? Before the construction of this example, in excess of 10^{15} games were simulated without generating a non-terminating game, so players can rest assured that the probability of encountering such a case "in the wild" are negligible.

Appendix A: Historical record of longest known terminating games. Note: these are records known to the authors and established by personal correspondence. Each record shows the starting packs, with cards on the left representing the top of each pack. The topmost player starts.

Mark Manasse (1992) 713 tricks, 5104 cards [**8**]

```
1. Q--J----K--K-J---Q---A---A
2. ---K-K--JA-QA--J-----Q----
```

Michael Kleber (before 1999). 805 tricks, 5791 cards [**1**]

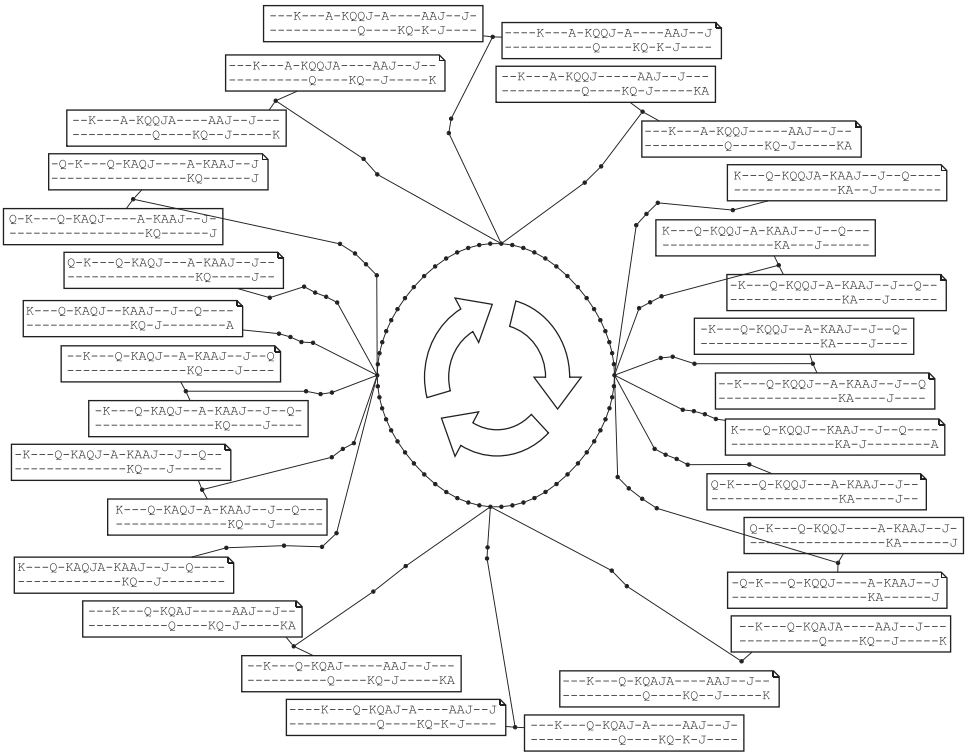

Figure 3. Illustration of a family of non-terminating games, all converging to the same 62-trick cycle. Starting packs shown in dogeared boxes are those with the topmost player starting; those in plain boxes start with the lower player.

```
1. ---JQ---K-A----A-J-K---QK-
2. -J-----------AJQA----K---Q
```

Michael Kleber (c. 2002). 841 tricks, 5977 cards

```
1. ----QJ-A-KK--------K--QQJ-
2. ---J---K--A-----Q-AJ-A----
```

Michael Kleber (c. 2005). 893 tricks, 6321 cards

```
1. -J--KA----A-Q--Q-A----KJ--
2. A------QKJ--Q-------KJ----
```

Truman Collins (2006). 960 tricks, 6914 cards

```
1. A-QK------Q----KA-----J---
2. -JAK----A--Q----J---QJ--K-
```

Richard Mann & Nicolas Wu (16-07-2007). 1007 tricks, 7157 cards

```
1. K-KK----K-A-----JAA--Q--J-
2. ---Q---Q-J-----J------AQ--
```

Reed Nessler (01-05-2012). 1015 tricks, 7207 cards

```
1. ----Q------A--K--A-A--QJK-
2. -Q--J--J---QK---K----JA---
```

Reed Nessler (04-05-2012). 1016 tricks, 7224 cards

```
1. -J-------Q------A--A--QKK-
2. -A-Q--J--J---Q--AJ-K---K--
```

Nicolas Wu (17-05-2012). 1016 tricks, 7225 cards

```
1. --A-Q--J--J---Q--AJ-K---K-
2. -J-------Q------A--A--QKK-
```

Reed Nessler (20-09-2013). 1014 tricks, 7259 cards

```
1. ---AK-Q--J----J--QKJ-Q----
2. ------JK-----A--K--Q---AA-
```

William Rucklidge (17-01-2014). 1024 tricks, 7269 cards

```
1. A-AQ-----Q--K--AQ-------JJ
2. -J-A-KKJ--K-----------Q---
```

Philip Anderson (03-02-2014). 1032 tricks, 7323 cards

```
1. -AJ--QK--K----Q--J-A-KKJ--
2. ---------JQ----------A-AQ-
```

William Rucklidge (05-03-2014). 1122 tricks, 7960 cards

```
1. -J------Q------AAA-----QQ-
2. K----JA-----------KQ-K-JJK
```

Reed Nessler (31-08-2021). 1106 tricks, 7972 cards

```
1. ----K---A--Q-A--JJA------J
2. -----KK---------A-JK-Q-Q-Q
```

Reed Nessler (09-06-2022). 1164 tricks, 8344 cards

```
1. ---AJ--Q---------QAKQJJ-QK
2. -----A----KJ-K--------A---
```

Appendix B: Example of games which lead to the same state after one trick. Note: the number in parentheses is the player who starts play.

Initial game:

```
1. --------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A-A-AJ- 2. J- (1)
```

Games after one trick of backwards play from the initial deck:

```
1. J--------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A-A-A 2. -J- (1)
1. J--------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A-A- 2. A-J- (2)
1. -J--------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A-A 2. A-J- (1)
1. AJ--------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A- 2. -A-J- (1)
1. -AJ--------------------J--Q-Q-Q-Q-K-KJ--K-K-A- 2. A-A-J- (2)
1. --AJ--------------------J--Q-Q-Q-Q-K-KJ--K-K-A 2. A-A-J- (1)
1. A-AJ--------------------J--Q-Q-Q-Q-K-KJ--K-K- 2. -A-A-J- (1)
1. A-AJ--------------------J--Q-Q-Q-Q-K-KJ--K-K 2. --A-A-J- (2)
```

```
1. -A-AJ--------------------J--Q-Q-Q-Q-K-KJ--K- 2. K-A-A-J- (2)
1. --A-AJ--------------------J--Q-Q-Q-Q-K-KJ--K 2. K-A-A-J- (1)
1. K-A-AJ--------------------J--Q-Q-Q-Q-K-KJ-- 2. -K-A-A-J- (1)
1. K-A-AJ--------------------J--Q-Q-Q-Q-K-KJ- 2. --K-A-A-J- (2)
1. -K-A-AJ--------------------J--Q-Q-Q-Q-K-KJ 2. --K-A-A-J- (1)
```

Appendix C: Found pieces and non-terminating standard games from them. Note: only pieces based on the definition provided are listed, there are thousands of strictly unique pieces.

Found pieces:

```
--
--Q-Q
-----K
--------
--K---Q-KQQ
--Q-Q-Q-Q-K-K
--K---A----AA
--------A-KAA
--A-----A-A--A
--A----Q----QQ
--------A----AA
--Q----Q----Q-Q
--Q--Q-Q-QQ-Q-Q
--------Q-Q-Q-Q
--------Q----Q-Q
--K-A--Q-K--Q--QQ
--Q----A-Q-A-A---
--------------K---
--------Q----Q----
-----------------Q-
--------------------
--------K----KQQ-KQQ
--------K----KK--Q-K--Q--QQ
--------K---------Q-Q-K---Q-KKQ
--------------------A----------A---A
--------------------Q----------Q----Q-Q
```

Constructed non-terminating standard games:

```
1. --------------------J--Q-Q-Q-Q-K-KJ--K-K-A-A-A-AJ- 2. J-
1. -----------------KJ--Q-K-A-K-Q-KJ--Q----A-Q-A-A-J- 2. J-
1. -----------------KJ--A-A-Q-A-A---Q-J--K---Q-KQK-J- 2. J-
1. --------Q----Q-Q-J--------A----AA---J--K-KQ-KK-AJ- 2. J-
1. --Q----Q----Q-Q-J--K---K-KAK--J--------A----AA--J- 2. J-
1. -----------------Q-J--K---Q-KQQJ--K-K--A-A-A--A-J- 2. J-
1. --------J--Q----A-Q-A-A-J--------K----KQ-Q-K-K-AJ- 2. J-
1. -----A--J--Q----Q----Q-QJ--------K----KA--A-KK-AJ- 2. J-
1. --------Q-------Q-J--Q-Q-K-K-AKJ--K---A----AA---J- 2. J-
1. --------Q-------Q-J--Q-Q-K-K-K-KJ--A-----A-A--A-J- 2. J-
```

```
1. --Q----Q----Q-Q-J-----KJ--------K----KA-A-K-A--AJ- 2. J-
1. --A-Q-K-K-K-KJ--Q----A-Q-A-AJ--------------Q----J- 2. J-
1. --------------Q----J--K---Q-KQKJ--A-A-K-A-A---Q-J- 2. J-
1. --K---A----AA---J--J--------K----KA--Q-K--Q--QQ-J- 2. J-
1. --A-A--J--A-A---J--------K---------Q-Q-K---Q-KKQJ- 2. J-
1. --K---A----AAJ--J--------K---------Q-Q-K---Q-KAQJ- 2. J-
```

Appendix D: A trace of the non-terminating cycle. Note: for each configuration, the number in parentheses is the player who starts play, i.e., the winner of the previous trick.

```
 1. 1. --K---A----AAJ--J--------K---------Q-Q-K---Q-KAQJ- 2. J- (1)
 2. 1. K---A----AAJ--J--------K---------Q-Q-K---Q-KAQJ- 2. --J- (2)
 3. 1. --A----AAJ--J--------K---------Q-Q-K---Q-KAQJ- 2. --K-J- (2)
 4. 1. ---AAJ--J--------K---------Q-Q-K---Q-KAQJ- 2. -----KA-J- (2)
 5. 1. --J--------K---------Q-Q-K---Q-KAQJ--------A-KAAJ- 2. J- (1)
 6. 1. J--------K---------Q-Q-K---Q-KAQJ--------A-KAAJ- 2. --J- (2)
 7. 1. --------K---------Q-Q-K---Q-KAQJ--------A-KAAJ--J- 2. J- (1)
 8. 1. ------K---------Q-Q-K---Q-KAQJ--------A-KAAJ--J- 2. --J- (2)
 9. 1. ---K---------Q-Q-K---Q-KAQJ--------A-KAAJ--J- 2. -----J- (2)
10. 1. --------Q-Q-K---Q-KAQJ--------A-KAAJ--J- 2. --------K-J- (2)
11. 1. Q-K---Q-KAQJ--------A-KAAJ--J- 2. ----------KQ-J- (2)
12. 1. -K---Q-KAQJ--------A-KAAJ--J--Q-- 2. ----------KQ-J- (1)
13. 1. ---Q-KAQJ--------A-KAAJ--J--Q----K--- 2. ----------KQ-J- (1)
14. 1. -KAQJ--------A-KAAJ--J--Q----K---------Q-- 2. -----KQ-J- (1)
15. 1. AQJ--------A-KAAJ--J--Q----K---------Q----K--- 2. -KQ-J- (1)
16. 1. --------A-KAAJ--J--Q----K---------Q----K---A-KQQJ- 2. J- (1)
17. 1. ------A-KAAJ--J--Q----K---------Q----K---A-KQQJ- 2. --J- (2)
18. 1. ---A-KAAJ--J--Q----K---------Q----K---A-KQQJ- 2. -----J- (2)
19. 1. KAAJ--J--Q----K---------Q----K---A-KQQJ- 2. --------A-J- (2)
20. 1. AAJ--J--Q----K---------Q----K---A-KQQJ--K--- 2. ----A-J- (1)
21. 1. AJ--J--Q----K---------Q----K---A-KQQJ--K---A---- 2. A-J- (1)
22. 1. --J--Q----K---------Q----K---A-KQQJ--K---A----AAJ- 2. J- (1)
23. 1. J--Q----K---------Q----K---A-KQQJ--K---A----AAJ- 2. --J- (2)
24. 1. --Q----K---------Q----K---A-KQQJ--K---A----AAJ--J- 2. J- (1)
25. 1. Q----K---------Q----K---A-KQQJ--K---A----AAJ--J- 2. --J- (2)
26. 1. ---K---------Q----K---A-KQQJ--K---A----AAJ--J- 2. --Q-J- (2)
27. 1. --------Q----K---A-KQQJ--K---A----AAJ--J- 2. -----Q-K-J- (2)
28. 1. -Q----K---A-KQQJ--K---A----AAJ--J- 2. -K-J----------Q-- (2)
29. 1. ---K---A-KQQJ--K---A----AAJ--J- 2. ----------Q----KQ-J- (2)
30. 1. ---A-KQQJ--K---A----AAJ--J--------K--- 2. ----Q----KQ-J- (1)
31. 1. QQJ--K---A----AAJ--J--------K---------A-Q-K--- 2. -KQ-J- (1)
32. 1. --K---A----AAJ--J--------K---------A-Q-K---Q-KQQJ- 2. J- (1)
33. 1. K---A----AAJ--J--------K---------A-Q-K---Q-KQQJ- 2. --J- (2)
34. 1. --A----AAJ--J--------K---------A-Q-K---Q-KQQJ- 2. --K-J- (2)
```

35. 1. ---AAJ--J--------K---------A-Q-K---Q-KQQJ- 2. -----KA-J- (2)
36. 1. --J--------K---------A-Q-K---Q-KQQJ--------A-KAAJ- 2. J- (1)
37. 1. J--------K---------A-Q-K---Q-KQQJ--------A-KAAJ- 2. --J- (2)
38. 1. --------K---------A-Q-K---Q-KQQJ--------A-KAAJ--J- 2. J- (1)
39. 1. ------K---------A-Q-K---Q-KQQJ--------A-KAAJ--J- 2. --J- (2)
40. 1. ---K---------A-Q-K---Q-KQQJ--------A-KAAJ--J- 2. -----J- (2)
41. 1. --------A-Q-K---Q-KQQJ--------A-KAAJ--J- 2. --------K-J- (2)
42. 1. Q-K---Q-KQQJ--------A-KAAJ--J- 2. ----------KA-J- (2)
43. 1. -K---Q-KQQJ--------A-KAAJ--J--Q-- 2. ----------KA-J- (1)
44. 1. ---Q-KQQJ--------A-KAAJ--J--Q----K--- 2. ----------KA-J- (1)
45. 1. -KQQJ--------A-KAAJ--J--Q----K---------Q-- 2. -----KA-J- (1)
46. 1. QQJ--------A-KAAJ--J--Q----K---------Q----K--- 2. -KA-J- (1)
47. 1. --------A-KAAJ--J--Q----K---------Q----K---Q-KQAJ- 2. J- (1)
48. 1. ------A-KAAJ--J--Q----K---------Q----K---Q-KQAJ- 2. --J- (2)
49. 1. ---A-KAAJ--J--Q----K---------Q----K---Q-KQAJ- 2. -----J- (2)
50. 1. KAAJ--J--Q----K---------Q----K---Q-KQAJ- 2. --------A-J- (2)
51. 1. AAJ--J--Q----K---------Q----K---Q-KQAJ--K--- 2. ----A-J- (1)
52. 1. AJ--J--Q----K---------Q----K---Q-KQAJ--K---A---- 2. A-J- (1)
53. 1. --J--Q----K---------Q----K---Q-KQAJ--K---A----AAJ- 2. J- (1)
54. 1. J--Q----K---------Q----K---Q-KQAJ--K---A----AAJ- 2. --J- (2)
55. 1. --Q----K---------Q----K---Q-KQAJ--K---A----AAJ--J- 2. J- (1)
56. 1. Q----K---------Q----K---Q-KQAJ--K---A----AAJ--J- 2. --J- (2)
57. 1. ---K---------Q----K---Q-KQAJ--K---A----AAJ--J- 2. --Q-J- (2)
58. 1. --------Q----K---Q-KQAJ--K---A----AAJ--J- 2. -----Q-K-J- (2)
59. 1. -Q----K---Q-KQAJ--K---A----AAJ--J- 2. -K-J----------Q-- (2)
60. 1. ---K---Q-KQAJ--K---A----AAJ--J- 2. ----------Q----KQ-J- (2)
61. 1. ---Q-KQAJ--K---A----AAJ--J--------K--- 2. ----Q----KQ-J- (1)
62. 1. QAJ--K---A----AAJ--J--------K---------Q-Q-K--- 2. -KQ-J- (1)

ACKNOWLEDGMENTS. We would like to dedicate this paper to the memory of John D. Beasley and John H. Conway who together did so much to popularize the question we have answered here. JDB also provided invaluable assistance in helping us to piece together the early history of the problem.

DISCLOSURE STATEMENT. No potential conflict of interest was reported by the authors.

ORCID
Richard P. Mann http://orcid.org/0000-0003-0701-1274

REFERENCES

[1] Paulhus MM. Beggar My Neighbour. Amer Math Monthly. 1999;106(2):162–165.
[2] Emerson M. Full list of Conway's anti-Hilbert questions?; 2023. Accessed: 2024-02-18. https://math.stackexchange.com/questions/4821876/full-list-of-conways-anti-hilbert-questions by Math Stack Exchange user May Emerson.
[3] Conway JH. Re: IMPatience; 1998. Private email to the math-fun mailing list run by Richard Schroeppel.
[4] Nowakowski R. More games of no chance. Cambridge: Cambridge University Press; 2002.

[5] Gowers T. Open problems concerning card games; 2008. Accessed: 2024-02-12. Available from: https://gowers.wordpress.com/2008/04/05/open-problems-concerning-card-games/.
[6] Beasley JD. The mathematics of games. Oxford: Oxford University Press; 1989.
[7] Long C. "Beggar Your Neighbor" has Infinite Loops - Maybe; 1991. Accessed: 2024-02-18. Usenet newsgroup post to rec.puzzles, archived at https://groups.google.com/g/rec.puzzles/c/ayJN4BR9QMM/m/CKBwi7yN0fIJ.
[8] Long C. The results of the Beggar Your Neighbor Contest; 1991. Accessed: 2024-02-18. Usenet newsgroup post to rec.puzzles, archived at https://groups.google.com/g/rec.puzzles/c/1CJlLrNmHvM/m/5v0rM0dML7cJ.
[9] Mazzaro D. Does this game terminate?. Accessed: 2024-10-28. Theoretical Computer Science Stack Exchange https://cstheory.stackexchange.com/q/52560.

BRAYDEN CASELLA received his B.S. in computer science from Rochester Institute of Technology in 2023
braydencasella@gmail.com

PHILIP M. ANDERSON received his M.Sci. in Chemistry in 2000 and his Ph.D. in Computational Chemistry in 2004, from Durham University.
p.m.anderson@dunelm.org.uk

MICHAEL KLEBER received his Ph.D. in mathematics from U.C. Berkeley in 1998, where getting math department computers to play BMN in their spare time was definitely not related to his research. He was a postdoctoral fellow at MIT and an assistant professor at Brandeis University. Since 2007 he has been a software engineer at Google.
michael.kleber@gmail.com

RICHARD P. MANN received his D. Phil. in Engineering Science from the University of Oxford in 2010. He is now Professor of Mathematical Ecology and Evolution at the University of Leeds.
Department of Statistics, School of Mathematics, University of Leeds
r.p.mann@leeds.ac.uk

REED NESSLER received his Ph.D. in mathematics from the University of Virginia in 2014. He is now a researcher at the Institute for Quantum Science and Engineering at Texas A&M University.
Institute for Quantum Science and Engineering, Texas A&M University
rcn@tamu.edu

WILLIAM RUCKLIDGE received his B.Sc. in Computer Science and Mathematics from the University of Toronto in 1987 and his Ph.D. in Computer Science from Cornell University in 1994. He has been at Google since 2006.
wjr@google.com

SAMUEL G. WILLIAMS is an independent researcher with a background in chemistry and mathematics. Currently working as a crypto miner, he is always looking for new problems to work on.
samwilliams98@gmail.com

NICOLAS WU received his D. Phil in Computer Science from the University of Oxford in 2010. He is now Professor of Computer Science at the Department of Computer Science, Imperial College London.
Department of Computer Science, Imperial College London
n.wu@imperial.ac.uk