n クイーン完成問題に対するカードベースゼロ知識証明

池田 昇太[1,a)]　品川 和雅[2,3,b)]

概要：n クイーン完成問題とは、いくつかのクイーンがあらかじめ配置された $n \times n$ のチェス盤において、互いに利き合わないようにすべての行にクイーンを配置できるかどうかを判定する問題である。この問題は、NP 完全問題の一つとして知られている。本稿では、この n クイーン完成問題に対して、カードベースゼロ知識証明プロトコルを 2 つ提案する。1 つ目は、パイルスクランブルシャッフルを 3 回用いる。2 つ目は、入力カード列の与え方を工夫することでシャッフルを 1 回削減し、パイルスクランブルシャッフルを 2 回用いるプロトコルである。

キーワード：カードベース暗号、ゼロ知識証明、n クイーン完成問題

Card-Based Zero-Knowledge Proofs for n-Queens Completion Problem

Shota Ikeda[1,a)]　Kazumasa Shinagawa[2,3,b)]

Abstract: The n-Queens Completion Problem is a decision problem in which, given an $n \times n$ chessboard with some queens already placed, one must determine whether it is possible to place queens on all rows so that they do not attack each other. This problem is known to be NP-complete. In this paper, we propose two card-based zero-knowledge proof protocols for the n-Queens Completion Problem. The first protocol uses a pile scramble shuffle three times. The second protocol reduces the number of shuffles by one by devising a method for providing the input card sequence, resulting in a total of two pile scramble shuffles.

Keywords: Card-based cryptography, Zero-Knowledge Proof, n-Queens Completion Problem

1. はじめに

1.1 n クイーン完成問題

n **クイーン問題**（n-Queens Problem）とは、$n \times n$ のチェス盤に n 個のチェスのクイーンを互いに利き合わないように配置する組合せ問題である。言い換えると、どの 2 つのクイーンも行・列・斜めが重ならないように配置する問題である。このパズルの起源は、1848 年にチェス愛好家であった Max Bezzel が、$n = 8$ の場合、すなわち一般的な 8×8 のチェス盤を用いた「8 クイーン問題」として提案したことに遡る。その 2 年後の 1850 年に、Franz Nauck [9] がこの問題を $n \times n$ の盤へと一般化し、今日知られる n クイーン問題の形となった。計算複雑性理論の観点から、この問題は「与えられた自然数 n に対し、解となるクイーンの配置は存在するか？」という決定問題として扱われる。$n \geq 4$ である全ての n に対して解が存在することが構成的に示されており、n クイーン問題は効率的に解を求めることが可能であるため、クラス P に属することが知られている。なお、$n = 2, 3$ の場合は明らかに解が存在しない。

一方、本稿で扱う n **クイーン完成問題**（n-Queens Completion Problem）は $n \times n$ のチェス盤にあらかじめいくつかのクイーンが互いに利き合わないように配置されており、「残りのクイーンを空いているマスに 1 つずつ配置し、最終的に n クイーン問題の解となる配置を完成させることが可能か？」という決定問題として定義される。この問題に関しても 1850 年に、Franz Nauck [9] が研究を行ってい

[1] 茨城大学 Ibaraki University
[2] 筑波大学 University of Tsukuba
[3] 産業技術総合研究所 National Institute of Advanced Industrial Science and Technology
[a)] 25nm709x@vc.ibaraki.ac.jp
[b)] shinagawa@cs.tsukuba.ac.jp

る。そして、Gent–Jefferson–Nightingale [5] によって n クイーン完成問題は NP 完全問題であることが示された。

1.2 カードベースゼロ知識証明

ゼロ知識証明とは、Goldwasser–Micali–Rackoff [6] によって提案された暗号技術である。これは、証明者 P と検証者 V の間で行われる対話型プロトコルであり、証明者はある問題に対して「問題に対する解を知っている」ことを検証者 V に納得させることができる一方で、その解自体についての情報を一切漏らさないという特徴を持つ。

ゼロ知識証明プロトコルは、次の 3 つの性質を満たす必要がある。

完全性 証明者 P が正しい解を知っている場合、検証者 V はその正当性を確信できる。

健全性 証明者 P が正しい解を知らない場合、どのようにごまかそうとしても、検証者 V はその不正を見抜くことができる。

ゼロ知識性 検証者 V は、証明を通じて正しい解に関するいかなる情報も得ることができない。

特に、「あるパズルに答えが存在する」という主張に対するゼロ知識証明は、パズルに対するゼロ知識証明と呼ばれる。これは、パズルの答えを知っている証明者 P が、答えを知らない検証者 V に対して、答えそのものを明かすことなく、そのパズルに解が存在することだけを納得させるプロトコルである。

さらに、カードや封筒などの身近な道具を使ってこのような証明を構成する研究も行われており、これらはカードベースゼロ知識証明と呼ばれる。カードベースゼロ知識証明は、ゼロ知識証明の概念を知らない非専門家にとっても直感的に理解しやすく、暗号技術への入り口として教育的な価値を有している。

1.3 カードベースゼロ知識証明の既存研究

これまでパズルに対するカードベースゼロ知識証明プロトコルは数多く提案されてきた。数独 [11,15] やカックロ [2,8]、マカロ [3,12]、15 パズル [14]、シャカシャカ [17]、ポリオミノパズル [16]、テントアンドツリー [18]、などでプロトコルが構成されている。

1.4 貢献

本稿では n クイーン完成問題に対するカードベースゼロ知識証明プロトコルを 2 つ提案する。提案プロトコル 1 はパイルスクランブルシャッフルを 3 回用いるプロトコルである。提案プロトコル 2 は入力カード列の与え方を工夫し、列の検証と斜線の検証を同時に行うことにより、パイルスクランブルシャッフルを提案プロトコル 1 より 1 回減らし、2 回用いるプロトコルである。

2. 準備

本章ではプロトコルで使用するカードとシャッフルについて解説する。さらに、n クイーン完成問題の厳密な定義を与える。

2.1 使用するカード

本稿ではカードベース暗号 [1,4] で主に用いられている 2 色カードと数字カードを使用する。2 色カードは黒 $\boxed{\clubsuit}$ と赤 $\boxed{\heartsuit}$ の 2 種類のカードを用いて、1 ビットの情報を $\boxed{\clubsuit} = 0$、$\boxed{\heartsuit} = 1$ として符号化する [10,13]。このとき、同じ色のカードは互いに区別できないものとする。この符号化に従って裏向きに置かれたカード $\boxed{?}$ をコミットメントと呼ぶ。

数字カードは $\boxed{1}\boxed{2}\cdots$ のように表面に数字が描かれ、裏面に $\boxed{?}$ が描かれているカードである。また、数字の上に線が入った数字カード $\boxed{\overline{1}}\boxed{\overline{2}}\cdots$ を**上線数字カード**と呼ぶ。同様に、数字の下に線が入った数字カード $\boxed{\underline{1}}\boxed{\underline{2}}\cdots$ を**下線数字カード**と呼ぶ。さらに、表面に何も描かれていないダミーカード $\boxed{}$ も用いる。これらのカードも 2 色カードと同様に裏向きに置かれたカードは区別できない。

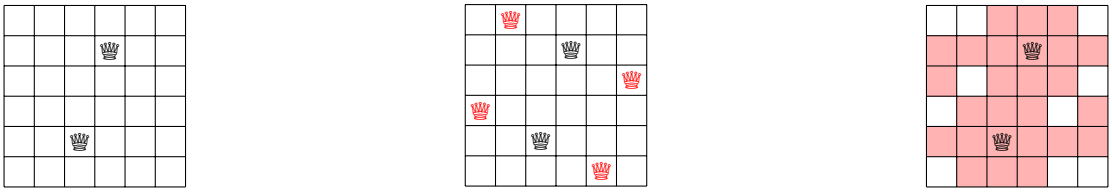
2.2 パイルスクランブルシャッフル（PSS）

パイルスクランブルシャッフル [7] とは、カード列をいくつかの等しいサイズの束に分割し、それぞれの束内のカードの順序はそのままに、束間の並び順だけを一様にランダムに入れ替える操作である。具体的には、n 個のカード束を $(p_1, p_2, \ldots, p_n)$ とし、それぞれの束 p_i は元のカード列から順に分けられた部分列である。このとき、シャッフル後のカード列は、一様ランダムに選ばれた置換 $\pi \in S_n$ に従って並べ替えられた $(p_{\pi^{-1}(1)}, p_{\pi^{-1}(2)}, \ldots, p_{\pi^{-1}(n)})$ となる。ここで、S_n は n 次対称群を表す。このシャッフル操作は $\left[\,\cdot \,\middle|\, \cdot \,\middle|\, \cdots \,\middle|\, \cdot\,\right]$ という表記を用いて表す。以下に例として、各 2 枚からなる計 3 個の束に対してパイルスクランブルシャッフルを適用した様子を示す。

$$\left[\begin{array}{c|c|c} \begin{smallmatrix}1&2\\?&?\end{smallmatrix} & \begin{smallmatrix}3&4\\?&?\end{smallmatrix} & \begin{smallmatrix}5&6\\?&?\end{smallmatrix} \end{array}\right] \rightarrow \left\{\begin{array}{l} \begin{smallmatrix}1&2&3&4&5&6\\?&?&?&?&?&?\end{smallmatrix} \\ \begin{smallmatrix}1&2&5&6&3&4\\?&?&?&?&?&?\end{smallmatrix} \\ \begin{smallmatrix}3&4&1&2&5&6\\?&?&?&?&?&?\end{smallmatrix} \\ \begin{smallmatrix}3&4&5&6&1&2\\?&?&?&?&?&?\end{smallmatrix} \\ \begin{smallmatrix}5&6&1&2&3&4\\?&?&?&?&?&?\end{smallmatrix} \\ \begin{smallmatrix}5&6&3&4&1&2\\?&?&?&?&?&?\end{smallmatrix} \end{array}\right.$$

2.3 問題の定義

本節ではカードベースゼロ知識証明プロトコルの構成にあたり、まず n クイーン完成問題の盤面を厳密に定義する。



(a) 例題　(b) 解答　(c) 禁止マス

図 1: 6 クイーン完成問題の例題・解答・禁止マス

また、各定義の意味を図 1 の具体例を用いて解説する。

盤面と座標　$n \times n$ のチェス盤を以下の集合 B として表す。

$$B = \{(i,j) | 1 \leq i,j \leq n\}$$

ここで (i,j) は i 行 j 列の座標を表す。

初期配置　初期配置として与えられるクイーンの集合を $Q_{\mathsf{pre}} \subset B$ とする。ただし、任意の異なるクイーン $(i_1,j_1),(i_2,j_2) \in Q_{\mathsf{pre}}$ は以下の条件をすべて満たす。

- $i_1 \neq i_2$（異なる行に属する）
- $j_1 \neq j_2$（異なる列に属する）
- $|i_1 - i_2| \neq |j_1 - j_2|$（異なる斜線に属する）

図 1 の例では $Q_{\mathsf{pre}} = \{(2,4),(5,3)\}$ となる。

禁止マス　初期配置クイーン Q_{pre} によって攻撃されているマス（初期配置クイーンの利きに入っているマス）を禁止マスとして定義する。証明者はこれらのマスに新たにクイーンを配置することはできない。禁止マスの集合 $S_{\mathsf{forbidden}}$ は、以下の式で定義される。

$$S_{\mathsf{forbidden}} := \{(i,j) \in B | \exists (i_q,j_q) \in Q_{\mathsf{pre}} \\ \text{s.t.}\, (i = i_q \vee j = j_q \vee |i - i_q| = |j - j_q|)\}$$

図 1 の例では、図 1c の赤いマスが禁止マスとなる。

配置候補マス　新たにクイーンを配置できる可能性のあるマスを配置候補マスとして定義する。これは、盤面の集合 B から禁止マスの集合 $S_{\mathsf{forbidden}}$ を除いた集合である。配置候補マスの集合 S_{cand} は以下の式で定まる。

$$S_{\mathsf{cand}} := B \setminus S_{\mathsf{forbidden}}$$

図 1 の例では図 1c の白いマスが配置候補マスとなる。

斜線グループ　任意の配置候補マス $(i,j) \in S_{\mathsf{cand}}$ に対し、そのマスと同じ斜線上に位置する他の配置候補マスの集合 $D(i,j)$ を以下のように定義する。

$$D(i,j) := \{(k,l) \in S_{\mathsf{cand}} | |i-k| = |j-l| \wedge (k,l) \neq (i,j)\}$$

図 1 の例では、配置候補マス $(1,2)$ の斜線グループ $D(1,2)$ は $\{(4,5)\}$ となり、配置候補マス $(3,2)$ の斜線グループ $D(3,2)$ は $\{(4,1),(6,5)\}$ となる。

正斜線グループ　任意の配置候補マス $(i,j) \in S_{\mathsf{cand}}$ に対し、そのマスと右下がり方向で同じ斜線上に位置する配置候補マスの集合 $D^+(i,j)$ を以下のように定める。

$$D^+(i,j) := \{(k,l) \in S_{\mathsf{cand}} | \, i - k = j - l\}$$

図 1 の例では、配置候補マス $(1,1)$ の正斜線グループ $D^+(1,1)$ は $\{(1,1),(6,6)\}$ となり、配置候補マス $(4,5)$ の正斜線グループ $D^+(4,5)$ は $\{(1,2),(4,5)\}$ となる。

負斜線グループ　任意の配置候補マス $(i,j) \in S_{\mathsf{cand}}$ に対し、そのマスと右上がり方向で同じ斜線上に位置する配置候補マスの集合 $D^-(i,j)$ を以下のように定める。

$$D^-(i,j) := \{(k,l) \in S_{\mathsf{cand}} | \, i - k = -(j - l)\}$$

図 1 の例では、配置候補マス $(1,6)$ の負斜線グループ $D^-(1,6)$ は $\{(1,6),(6,1)\}$ となり、配置候補マス $(4,5)$ の負斜線グループ $D^-(4,5)$ は $\{(3,6),(4,5)\}$ となる。

この定義により、初期配置の制約下で、原理的にクイーンを配置し得るすべてのマスが明確に特定される。提案プロトコルでは、証明者はこの S_{cand} の部分集合として解 S_{ans} を構成し、その妥当性を検証者に証明することになる。

3. 提案プロトコル

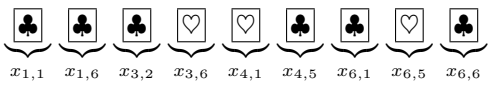
本章では提案プロトコル 1 と提案プロトコル 2 の手順と使用するカード枚数、正当性についてそれぞれ解説する。

3.1 提案プロトコル 1

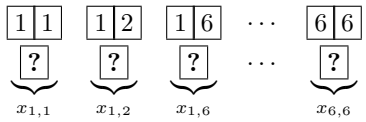
3.1.1 提案プロトコル 1 の手順

本節では、提案プロトコル 1 の手順を解説する。証明者 P は、自身の知る解（クイーンを配置するマスの集合 $S_{\mathsf{ans}} \subset S_{\mathsf{cand}}$）に基づき、以下の手順を実行する。また、手順の解説を分かりやすくするため、図 1 の問題に対して提案プロトコル 1 を実行した際の様子を示しながら解説する。

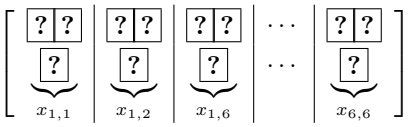
(1) **入力:** 証明者 P は、自身の解 S_{ans} に対応するコミットメントとして、各配置候補マス $(i,j) \in S_{\mathsf{cand}}$ にコミットメントを裏向きで配置する。このとき、$(i,j) \in S_{\mathsf{ans}}$ であれば $\boxed{\heartsuit}$ を、そうでなければ $\boxed{\clubsuit}$ を配置し、検証者に秘匿して入力する。例として、図 1 で証明者 P が正しい解に従ってコミットメントを入力した場合、以下のように配置される。



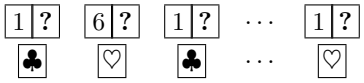
(2) **ラベリング:** 裏向きに配置されたコミットメントの上に、その座標を示す行の数字カード$\boxed{i}$と列の数字カード$\boxed{j}$を表向きで置く。例として、図1では以下のように配置される。



(3) **行制約検証:** すべての数字カードを裏向きにした上で、各マス上のカード束（裏向きのコミットメントおよび2枚の数字カード）を1つの束にして、パイルスクランブルシャッフルを実行する。

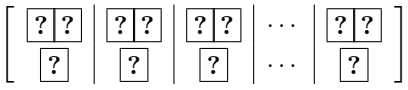


シャッフル後、すべてのコミットメントと**行の数字カード**を表にする。

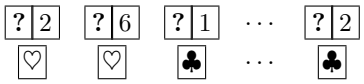


このとき、初期配置クイーンが置かれていない各行について$\boxed{\heartsuit}$のカードがちょうど1枚のみ存在することを確認する。すべての行で条件を満たす場合は次の手順に進み、そうでなければ検証者Vは証明を棄却する。

(4) **列制約検証:** すべてのコミットメントと行の数字カードを裏向きにし、再度パイルスクランブルシャッフルを実行する。

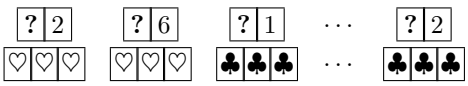


シャッフル後、すべてのコミットメントと**列の数字カード**を表にする。

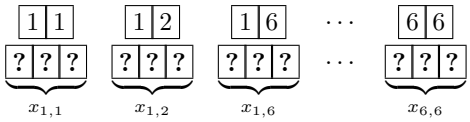


このとき、初期配置クイーンがいない各列について$\boxed{\heartsuit}$のカードがちょうど1枚のみ存在することを確認する。すべての列で条件を満たす場合次の手順に進み、そうでなければ検証者Vは証明を棄却する。

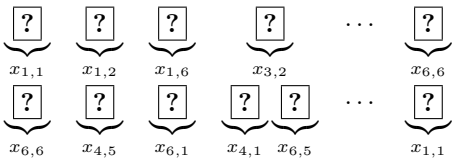
(5) **カードのコピー:** 各コミットメントで同一のカードをN_{copy}枚準備し、表向きで追加する。N_{copy}は、盤面上の配置候補マス$(i,j) \in S_{\mathsf{cand}}$が持ちうる斜線グループ$D(i,j)$の要素数の最大値、すなわち$N_{\mathsf{copy}} := \max_{(i,j)\in S_{\mathsf{cand}}} |D(i,j)|$として定義される。図1の例では、$N_{\mathsf{copy}} = 2$となるため、各コミットメントで2枚のカードを追加して以下のようになる。



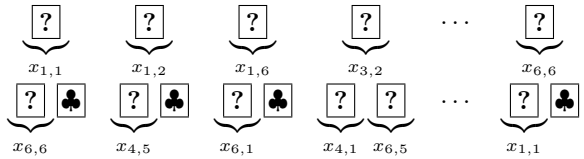
(6) **斜線グループの形成とパディング:** すべてのコミットメント（オリジナルおよびコピー）と列の数字カードを裏向きにし、再度パイルスクランブルシャッフルした後、行の数字カードと列の数字カードを全て表にし、カード束を整列させる。



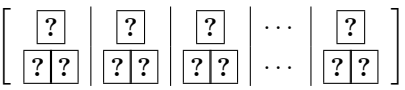
その後、各コミットメントの行と列の数字カードをすべて取り除く。次に各コミットメント$x_{i,j}$に対し、その斜線グループ$D(i,j)$に属するすべてのコミットメントを$x_{i,j}$の下に集めることで、斜線グループを形成する。図1の例では以下のようになる。



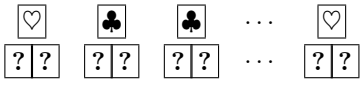
また情報漏洩を防ぐため、各斜線グループのカード枚数（コミットメント$x_{i,j}$の下に集めたカード枚数）が最大サイズ（N_{copy}）に達するまで、不足分のカードを$\boxed{\clubsuit}$追加することで補う。図1の例では以下のようになる。



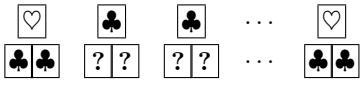
(7) **斜線制約検証:** 追加した$\boxed{\clubsuit}$をすべて裏にする。形成した各斜線グループ内で束を作り、パイルスクランブルシャッフルを実行する。



その後、各グループにおいて、上段にあるコミットメント$x_{i,j}$をすべて表にする。



このとき、表にした$x_{i,j}$が$\boxed{\heartsuit}$であるグループに対して、その下にある2色カードをすべて表にする。



このとき、下段にあるすべての2色カードが$\boxed{\clubsuit}$であることを確認する。この条件が満たされた場合、検証者Vは証明を受理し、そうでなければ証明を棄却する。

3.1.2 提案プロトコル 1 のカード枚数

本節では、提案プロトコル 1 で用いる 2 色カード枚数と数字カードのカード枚数について解説する。

2 色カード 手順（1）の**入力**で、$|S_{\mathsf{cand}}|$ 枚の 2 色カードを入力する。図 1 の例では、入力として 10 枚のコミットメントが入力される。また、手順（5）の**カードのコピー**では各コミットメントごとに N_{copy} 枚コピーをするため、$N_{\mathsf{copy}} \times |S_{\mathsf{cand}}|$ 枚追加することになる。図 1 の例では、各コミットメントごとに 2 枚の 2 色カードを追加することになるので、合計 20 枚追加することになる。そして、手順（6）の**斜線グループの形成とパディング**でダミーとして $\boxed{\clubsuit}$ を $\sum_{(i,j)\in S_{\mathsf{cand}}}(N_{\mathsf{copy}} - |D(i,j)|)$ 枚追加することになる。図 1 の例では、8 枚の $\boxed{\clubsuit}$ をダミーとして追加することになる。よって、提案プロトコル 1 では 2 色カードを合計 $(N_{\mathsf{copy}}+1) \times |S_{\mathsf{cand}}| + \sum_{(i,j)\in S_{\mathsf{cand}}}(N_{\mathsf{copy}} - |D(i,j)|)$ 枚使用する。図 1 の例では、2 色カードを合計 38 枚使用することになる。

数字カード 手順（2）の**ラベリング**にて各行の数字カードを $|S_{\mathsf{cand}}|$ 枚、各列の数字カードを $|S_{\mathsf{cand}}|$ 枚用いることになる。よって、合計 $2|S_{\mathsf{cand}}|$ 枚の数字カードを使用することになる。また、図 1 の例では各行で 10 枚（$\boxed{1}$ が 3 枚、$\boxed{3}$ が 2 枚、$\boxed{4}$ が 2 枚、$\boxed{6}$ が 3 枚）使用し、各列で 10 枚（$\boxed{1}$ が 3 枚、$\boxed{2}$ が 2 枚、$\boxed{5}$ が 2 枚、$\boxed{6}$ が 3 枚）使用することになる。

3.1.3 提案プロトコル 1 の正当性

本節では、提案プロトコル 1 が完全性、健全性、ゼロ知識性を満たしていることを確認する。

完全性 証明者 P が n クイーン完成問題に対する正しい解を知っており、その解に従って忠実にプロトコルを実行すると仮定する。このプロトコルでは、正しい解が持つ性質をそれぞれの手順で検証している。このパズルは各行に必ずクイーンが 1 つ存在する。これは、手順（3）の**行制約検証**で確認される。同様に、各列に必ずクイーンが 1 つ存在する。これは、手順（4）の**列制約検証**で確認される。さらに、このパズルではクイーンが同じ斜線上に存在しない。これは、手順（7）の**斜線制約検証**で確認される。以上のように、証明者が正しい解に従う場合、すべての検証を通過するため、完全性が満たされていることがわかる。

健全性 証明者 P が n クイーン完成問題に対する正しい解を持たずに、検証者 V を欺こうとする場合を考える。このとき、証明のいずれかの段階で不正が必ず露見し、証明が棄却されることを示す。証明者 P の不正は、「ある行にクイーンが複数ある（ある行にクイーンが 1 つもない）」場合、「ある列にクイーンが複数ある（ある列にクイーンが 1 つもない）」場合、「ある斜線上にクイーンが複数ある」のいずれかに分類される。まず、「ある行にクイーンが複数ある（ある行にクイーンが 1 つもない）」場合は手順（3）の**行制約検証**で矛盾が発覚する。次に、「ある列にクイーンが複数ある（ある列にクイーンが 1 つもない）」場合は手順（4）の**列制約検証**で矛盾が発覚する。そして、「ある斜線上にクイーンが複数ある」場合は手順（7）の**斜線制約検証**で矛盾が発覚する。以上より、証明者が正しい解を持たない場合、その不正が「ある行にクイーンが複数ある（ある行にクイーンが 1 つもない）」場合、「ある列にクイーンが複数ある（ある列にクイーンが 1 つもない）」場合、「ある斜線上にクイーンが複数ある」どれであっても、プロトコルのいずれかの検証段階で必ず検出されるため、健全性が満たされていることがわかる。

ゼロ知識性 提案プロトコルでは正しい解に従ってカードが配置されていた場合、各行と各列にクイーンが 1 つあることになる。このとき、手順（3）と（4）のパイルスクランブルシャッフル（2.2 節）により、その並び順は一様ランダムになるため、各行・各列のどこにクイーンがあるかの情報は一切漏れない。また、手順（7）でもパイルスクランブルシャッフル（2.2 節）により、その並び順は一様ランダムになるため、上段のコミットメントをすべて表にしてもクイーンの位置に関する情報は一切漏れない。さらに、表にしたコミットメントが $\boxed{\heartsuit}$ だったものの下にあるコミットメントをすべて表にしており、正しい解に従ってカードが配置されていれば、すべて $\boxed{\clubsuit}$ がめくられるはずであり、クイーンの位置に関する情報は漏れない。よって、ゼロ知識性が満たされていることがわかる。

3.2 提案プロトコル 2

3.2.1 提案プロトコル 2 の手順

本節では、提案プロトコル 2 の手順を解説する。証明者 P は、自身の知る解（クイーンを配置するマスの集合 $S_{\mathsf{ans}} \subset S_{\mathsf{cand}}$）に基づき、以下の手順を実行する。また、手順のイメージを分かりやすくするため、図 1 の問題に対して提案プロトコル 2 を実行した際の様子を示しながら解説する。

このプロトコルでは、正斜線グループと負斜線グループとして、要素数 2 以上のもののみ扱う。正斜線グループそれぞれに対して番号を割り振る。図 1 の例では正斜線グループ $\{(1,1),(6,6)\}$ には 1、$\{(1,2),(4,5)\}$ には 2、$\{(3,2),(6,5)\}$ には 3 を割り振る。同様に、負斜線グループにも番号を割り振る。$\{(1,6),(6,1)\}$ には 1、$\{(3,2),(4,1)\}$ には 2、$\{(3,6),(4,5)\}$ には 3 を割り振る。さらに、このプロトコルでは証明者 P が持っている解の知識を用いる。

(1) **入力:** 証明者 P は、自身の解 S_{ans} に対応するコミットメントとして、各配置候補マス $(i,j) \in S_{\mathsf{cand}}$ にコミッ

トメントを裏向きで配置する。このとき、$(i,j) \in S_{\mathsf{ans}}$ であれば $\boxed{\heartsuit}$ を、そうでなければ $\boxed{\clubsuit}$ を配置し、検証者に秘匿して入力する。

また、後の斜線グループの検証を行う際の情報漏洩を防ぐため[*1]、 証明者 P は解の知識を用いてダミーとなるカード束を検証者 V に秘匿していくつか作成する。ある正斜線グループにおいて、クイーンが含まれない、つまり $D^+(i,j) \cap S_{\mathsf{ans}} = \emptyset$ となる正斜線グループ k に対して以下のカード束を 1 個追加する。

$$\begin{array}{|c|}\hline \overline{k} \\ \hline \heartsuit \\ \hline \end{array}$$

一方、ある正斜線グループにおいて、クイーンが含まれる、つまり $D^+(i,j) \cap S_{\mathsf{ans}} \neq \emptyset$ となる正斜線グループ k に対して以下のカード束を 1 個追加する。

$$\begin{array}{|c|}\hline \overline{k} \\ \hline \clubsuit \\ \hline \end{array}$$

これを負斜線グループに対しても行う。ただし、負斜線グループでは、下線数字カードを用いる。図 1 の例では、証明者 P は以下のカード束を追加する。

$$\begin{array}{|c|c|c|c|c|c|}\hline \underline{\overline{1}} & \underline{\overline{2}} & \underline{\overline{3}} & \underline{1} & \underline{2} & \underline{3} \\ \hline \heartsuit & \clubsuit & \clubsuit & \heartsuit & \clubsuit & \clubsuit \\ \hline \end{array}$$

なお、これらのカード束は手順（5）から用いる。

(2) **ラベリング:** 裏向きに配置されたコミットメントの上に、その座標を示す行の数字カード $\boxed{i}$ と列の数字カード $\boxed{j}$ を表向きで置く。さらに、各コミットメント（配置候補マス）が属する正斜線グループの番号 k の上線数字カード $\boxed{\overline{k}}$ を表向きで置く。同様に、負斜線グループの番号 l の下線数字カード $\boxed{\underline{l}}$ を表向きでおく。このとき、正斜線グループのみ、または負斜線グループのみに属するようなコミットメント（配置候補マス）では、属していないグループにはダミーカード $\boxed{\ }$ を置くようにする。例として、図 1 では以下のように配置される。

$$\begin{array}{|c|c|c|c|}\hline 1 & 1 & \overline{1} & \ \ \\ \hline \multicolumn{3}{c}{\underbrace{\boxed{?}}_{x_{1,1}}} \end{array} \quad \begin{array}{|c|c|c|c|}\hline 1 & 2 & \overline{2} & \ \ \\ \hline \multicolumn{3}{c}{\underbrace{\boxed{?}}_{x_{1,2}}} \end{array} \quad \begin{array}{|c|c|c|c|}\hline 1 & 6 & \ \ & \underline{4} \\ \hline \multicolumn{3}{c}{\underbrace{\boxed{?}}_{x_{1,6}}} \end{array} \quad \cdots \quad \begin{array}{|c|c|c|c|}\hline 6 & 6 & \overline{1} & \ \ \\ \hline \multicolumn{3}{c}{\underbrace{\boxed{?}}_{x_{6,6}}} \end{array}$$

(3) **行制約検証:** すべての数字カードと上線数字カード、下線数字カードを裏向きにし、各マス上のカード列を 1 つの束にして、パイルスクランブルシャッフルを実行する。

$$\left[\ \begin{array}{|c|c|c|c|}\hline ? & ? & ? & ? \\ \hline \multicolumn{3}{c}{\underbrace{\boxed{?}}_{x_{1,1}}} \end{array} \ \middle|\ \begin{array}{|c|c|c|c|}\hline ? & ? & ? & ? \\ \hline \multicolumn{3}{c}{\underbrace{\boxed{?}}_{x_{1,2}}} \end{array} \ \middle|\ \cdots \ \middle|\ \begin{array}{|c|c|c|c|}\hline ? & ? & ? & ? \\ \hline \multicolumn{3}{c}{\underbrace{\boxed{?}}_{x_{6,6}}} \end{array} \ \right]$$

シャッフル後、すべてのコミットメントと**行の数字カード**を表にする。

$$\begin{array}{|c|c|c|c|}\hline 3 & ? & ? & ? \\ \hline \clubsuit \\ \cline{1-1} \end{array} \quad \begin{array}{|c|c|c|c|}\hline 1 & ? & ? & ? \\ \hline \heartsuit \\ \cline{1-1} \end{array} \quad \cdots \quad \begin{array}{|c|c|c|c|}\hline 3 & ? & ? & ? \\ \hline \heartsuit \\ \cline{1-1} \end{array}$$

このとき、初期配置クイーンが置かれていない各行について $\boxed{\heartsuit}$ のカードがちょうど 1 枚のみ存在することを確認する。すべての行で条件を満たす場合は次の手順に進み、そうでなければ検証者 V は証明を棄却する。

(4) **カードのコピー:** 各コミットメントで同一のカードを 2 枚準備し、表向きで追加する。図 1 の例では、以下のようになる。

$$\begin{array}{|c|c|c|c|}\hline 3 & ? & ? & ? \\ \hline \clubsuit & \clubsuit & \clubsuit \\ \cline{1-3} \end{array} \quad \begin{array}{|c|c|c|c|}\hline 1 & ? & ? & ? \\ \hline \heartsuit & \heartsuit & \heartsuit \\ \cline{1-3} \end{array} \quad \cdots \quad \begin{array}{|c|c|c|c|}\hline 3 & ? & ? & ? \\ \hline \heartsuit & \heartsuit & \heartsuit \\ \cline{1-3} \end{array}$$

その後、表にした行の数字カードをすべて取り除く。

$$\begin{array}{|c|c|c|}\hline ? & ? & ? \\ \hline \clubsuit & \clubsuit & \clubsuit \\ \cline{1-3} \end{array} \quad \begin{array}{|c|c|c|}\hline ? & ? & ? \\ \hline \heartsuit & \heartsuit & \heartsuit \\ \cline{1-3} \end{array} \quad \cdots \quad \begin{array}{|c|c|c|}\hline ? & ? & ? \\ \hline \heartsuit & \heartsuit & \heartsuit \\ \cline{1-3} \end{array}$$

(5) **列制約検証と斜線制約検証:** 手順（4）後のカード列に、手順（1）で作成したカード束を追加する。

$$\underbrace{\begin{array}{|c|c|c|}\hline ? & ? & ? \\ \hline \clubsuit & \clubsuit & \clubsuit \\ \cline{1-3} \end{array} \quad \begin{array}{|c|c|c|}\hline ? & ? & ? \\ \hline \heartsuit & \heartsuit & \heartsuit \\ \cline{1-3} \end{array} \quad \cdots \quad \begin{array}{|c|c|c|}\hline ? & ? & ? \\ \hline \heartsuit & \heartsuit & \heartsuit \\ \cline{1-3} \end{array}}_{\text{手順（4）後のカード束}} \qquad \underbrace{\begin{array}{|c|c|c|c|}\hline ? & ? & \cdots & ? \\ \hline ? & ? & \cdots & ? \\ \cline{1-4} \end{array}}_{\text{手順（1）で作成したカード束}}$$

その後、表面になっている 2 色カードをすべて裏にして、以下のパイルスクランブルシャッフルを実行する。

$$\left[\ \begin{array}{|c|}\hline ? \\ \hline ? \\ \hline \end{array} \ \middle|\ \begin{array}{|c|}\hline ? \\ \hline ? \\ \hline \end{array} \ \middle|\ \begin{array}{|c|}\hline ? \\ \hline ? \\ \hline \end{array} \ \middle|\ \cdots \ \middle|\ \begin{array}{|c|}\hline ? \\ \hline ? \\ \hline \end{array} \ \right]$$

シャッフル後、上段にあるすべてのカードを表にする。

$$\begin{array}{|c|c|c|c|c|}\hline 2 & \underline{3} & \ \ & \cdots & \underline{\overline{1}} \\ \hline ? & ? & ? & \cdots & ? \\ \hline \end{array}$$

このとき、めくられたカードがダミーカード $\boxed{\ }$ でなかったカードに対して、その下にあるカードをすべて表にする。

$$\begin{array}{|c|c|c|c|c|}\hline 2 & \underline{3} & \ \ & \cdots & \underline{\overline{1}} \\ \hline \clubsuit & \heartsuit & ? & \cdots & \heartsuit \\ \hline \end{array}$$

このとき、以下をすべて確認する。

- 初期配置クイーンがいない各列について、$\boxed{\heartsuit}$ のカードが 1 枚のみ存在することを確認する。
- 各正斜線グループについて、$\boxed{\heartsuit}$ のカードが 1 枚のみ存在することを確認する。
- 各負斜線グループについて、$\boxed{\heartsuit}$ のカードが 1 枚のみ存在することを確認する。

以上のすべての条件を満たすとき検証者 V は証明を受理し、そうでなければ証明を棄却する。

[*1] ある正斜線グループまたは負斜線グループにクイーンが存在しないという情報を漏らさないため。

3.2.2 提案プロトコル 2 のカード枚数

本節では、提案プロトコル 2 で用いる 2 色カード枚数と数字カード、上線数字カード、下線数字カード、ダミーカードのカード枚数について解説する。このとき、正斜線グループの個数を N^+ とし、負斜線グループの個数を N^- と定義する。また、各正斜線グループのサイズを $|D_k^+|$ とし、各負斜線グループのグループのサイズを $|D_l^-|$ と定義する。

2 色カード　手順（1）の**入力**で、$|S_{\mathsf{cand}}|$ 枚、ダミーとなるカード束で $N^+ + N^-$ 枚用いる。また、手順（4）の**カードのコピー**にて $2|S_{\mathsf{cand}}|$ 枚用いる。よって、提案プロトコル 2 で使用するカード枚数は $3|S_{\mathsf{cand}}| + N^+ + N^-$ 枚となる。図 1 の例では 36 枚となる。

数字カード　手順（2）の**ラベリング**にて各行の数字カードを $|S_{\mathsf{cand}}|$ 枚、各列の数字カードを $|S_{\mathsf{cand}}|$ 枚用いることになる。よって、合計 $2|S_{\mathsf{cand}}|$ 枚の数字カードを使用することになる。また、図 1 の例では各行で 10 枚（$\boxed{1}$ が 3 枚、$\boxed{3}$ が 2 枚、$\boxed{4}$ が 2 枚、$\boxed{6}$ が 3 枚）使用し、各列で 10 枚（$\boxed{1}$ が 3 枚、$\boxed{2}$ が 2 枚、$\boxed{5}$ が 2 枚、$\boxed{6}$ が 3 枚）使用することになる。

上線数字カード　手順（1）の**入力**にて、ダミーとなるカード束として N^+ 枚用いる。また、手順（2）にて $\sum_{k=1}^{N^+} |D_k^+|$ 枚用いる。よって、合計 $\sum_{k=1}^{N^+} |D_k^+| + N^+$ 枚となる。図 1 の例では、$N^+ = 3$ となり、上線数字カードは 9 枚用いる。

下線数字カード　手順（1）の**入力**にて、ダミーとなるカード束として N^- 枚用いる。また、手順（2）にて $\sum_{l=1}^{N^-} |D_l^-|$ 枚用いる。よって、合計 $\sum_{l=1}^{N^-} |D_l^-| + N^-$ 枚となる。図 1 の例では、$N^- = 3$ となり、上線数字カードは 9 枚用いる。

ダミーカード　ダミーカードのカード枚数は $2|S_{\mathsf{cand}}| - (\sum_{k=1}^{N^+} |D_k^+| + \sum_{l=1}^{N^-} |D_l^-|)$ 枚となる。図 1 の例では、8 枚となる。

3.2.3 提案プロトコル 2 の正当性

本節では、提案プロトコル 2 が完全性、健全性、ゼロ知識性を満たしていることを確認する。

完全性　証明者 P が n クイーン完成問題に対する正しい解を知っており、その解に従って忠実にプロトコルを実行すると仮定する。このプロトコルでは、正しい解が持つ性質をそれぞれの手順で検証している。このパズルは各行に必ずクイーンが 1 つ存在する。これは、手順（3）の**行制約検証**で確認される。同様に、各列に必ずクイーンが 1 つ存在する。これは、手順（5）の**列制約検証と斜線制約検証**で確認される。さらに、このパズルではクイーンが同じ斜線上に存在しない。これも、手順（5）の**列制約検証と斜線制約検証**で確認される。以上のように、証明者が正しい解に従う場合、すべての検証を通過するため、完全性が満たされていることがわかる。

健全性　証明者 P が n クイーン完成問題に対する正しい解を持たずに、検証者 V を欺こうとする場合を考える。このとき、証明のいずれかの段階で不正が必ず露見し、証明が棄却されることを示す。証明者 P の不正は、「ある行にクイーンが複数ある（ある行にクイーンが 1 つもない）」場合、「ある列にクイーンが複数ある（ある列にクイーンが 1 つもない）」場合、「ある斜線上にクイーンが複数ある」のいずれかに分類される。まず、「ある行にクイーンが複数ある（ある行にクイーンが 1 つもない）」場合は手順（3）の**行制約検証**で矛盾が発覚する。次に、「ある列にクイーンが複数ある（ある列にクイーンが 1 つもない）」場合は手順（5）の**列制約検証と斜線制約検証**で矛盾が発覚する。そして、「ある斜線上にクイーンが複数ある」場合も手順（5）の**列制約検証と斜線制約検証**で矛盾が発覚する。以上より、証明者が正しい解を持たない場合、その不正が「ある行にクイーンが複数ある（ある行にクイーンが 1 つもない）」場合、「ある列にクイーンが複数ある（ある列にクイーンが 1 つもない）」場合、「ある斜線上にクイーンが複数ある」どれであっても、プロトコルのいずれかの検証段階で必ず検出されるため、健全性が満たされていることがわかる。

ゼロ知識性　提案プロトコルでは正しい解に従ってカードが配置されていた場合、各行と各列にクイーンが 1 つあることになる。このとき、手順（2）にて各コミットメントに対して列・行・正斜線グループ・負斜線グループそれぞれを表す数字カードを置き、そのコミットメントが属していないグループにはダミーカードを置くことで、各コミットメントの束のカード枚数が 5 枚になるため、束の識別はできない。その上で、手順（3）のパイルスクランブルシャッフル（2.2 節）により、その並び順は一様ランダムになるため、各行のどこにクイーンがあるかの情報は一切漏れない。また、列の検証でも同様に各列のどこにクイーンがあるかの情報は一切漏れない。それに、手順（1）でダミーとなるカード束を各正斜線グループおよび各負斜線グループに 1 つ追加している。そのため、ある正斜線グループまたは負斜線グループにクイーンが存在しないという情報は漏れない。さらに、手順（5）のパイルスクランブルシャッフル（2.2 節）により、その並び順は一様ランダムになるため、クイーンの位置に関する情報は一切漏れない。よって、ゼロ知識性が満たされていることがわかる。

4. おわりに

本稿では、NP 完全問題として知られる n クイーン完成問題に対するカードベースゼロ知識証明プロトコルとして

パイルスクランブルシャッフルが定数回（3 回と 2 回）のプロトコルを提案した。今後の課題としては、プロトコルの効率化としてシャッフル回数やカード枚数の削減などが挙げられる。

謝辞　本研究は JSPS 科研費 JP23H00479、JP21K17702 と JST CREST JPMJCR22M1 の支援を受けた。

参考文献

[1] B. D. Boer. More efficient match-making and satisfiability the five card trick. In J.-J. Quisquater and J. Vandewalle eds., *EUROCRYPT 1989*, Vol. 434 of *LNCS*, pp. 208–217, Heidelberg, 1990. Springer.

[2] X. Bultel, J. Dreier, J. Dumas, and P. Lafourcade. Physical zero-knowledge proofs for akari, takuzu, kakuro and kenken. In *8th International Conference on Fun with Algorithms, FUN 2016*, Vol. 49 of *LIPIcs*, pp. 8:1–8:20, 2016.

[3] X. Bultel, J. Dreier, J. Dumas, P. Lafourcade, D. Miyahara, T. Mizuki, A. Nagao, T. Sasaki, K. Shinagawa, and H. Sone. Physical zero-knowledge proof for makaro. In *Stabilization, Safety, and Security of Distributed Systems - 20th International Symposium, SSS 2018*, Vol. 11201 of *Lecture Notes in Computer Science*, pp. 111–125, 2018.

[4] C. Crépeau and J. Kilian. Discreet solitary games. In D. R. Stinson ed., *Advances in Cryptology—CRYPTO' 93*, Vol. 773 of *LNCS*, pp. 319–330, Berlin, Heidelberg, 1994. Springer.

[5] I. P. Gent, C. Jefferson, and P. Nightingale. Complexity of n-queens completion. *Journal of Artificial Intelligence Research*, 59:815–848, 2017.

[6] S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof-systems (extended abstract). In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing*, pp. 291–304, 1985.

[7] R. Ishikawa, E. Chida, and T. Mizuki. Efficient card-based protocols for generating a hidden random permutation without fixed points. In C. S. Calude and M. J. Dinneen eds., *Unconventional Computation and Natural Computation*, Vol. 9252 of *LNCS*, pp. 215–226, Cham, 2015. Springer.

[8] D. Miyahara, T. Sasaki, T. Mizuki, and H. Sone. Card-based physical zero-knowledge proof for kakuro. *IEICE Transactions*, 102-A(9):1072–1078, 2019.

[9] F. Nauck. Schach: Eine in das gebiet der mathematik fallende aufgabe von herrn dr. nauck in schleusingen. *Illustrirte Zeitung*, 14(361):352, 1850. Cited in Campbell (1977).

[10] V. Niemi and A. Renvall. Secure multiparty computations without computers. *Theor. Comput. Sci.*, 191(1–2):173–183, 1998.

[11] T. Ono, S. Ruangwises, Y. Abe, K. Hatsugai, and M. Iwamoto. Single-shuffle physical zero-knowledge proof for sudoku using interactive inputs. In *Proceedings of the 12th ACM ASIA Public-Key Cryptography Workshop*, pp. 1–8, 2025.

[12] S. Ruangwises and T. Itoh. Physical ZKP for makaro using a standard deck of cards. *CoRR*, abs/2112.12042, 2021.

[13] K. Shinagawa. Card-based protocols with single-card encoding. In C. Anutariya and M. M. Bonsangue eds., *Theoretical Aspects of Computing*, Vol. 15373 of *LNCS*, pp. 182–194, Cham, 2025. Springer.

[14] Y. Tamura, A. Suzuki, and T. Mizuki. Card-based zero-knowledge proof protocols for the 15-puzzle and the token swapping problem. In *Proceedings of the 11th ACM Asia Public-Key Cryptography Workshop*, pp. 11–22, 2024.

[15] K. Tanaka, S. Sasaki, K. Shinagawa, and T. Mizuki. Only two shuffles perform card-based zero-knowledge proof for sudoku of any size. In *2025 Symposium on Simplicity in Algorithms (SOSA)*, pp. 94–107. SIAM, 2025.

[16] 室大地, 小泉康一, 千田栄幸, 水木敬明. ポリオミノパズルに対するカードを用いたゼロ知識証明プロトコルの改良. 第 110 回コンピュータセキュリティ・第 60 回セキュリティ心理学とトラスト合同研究発表会, 2025.

[17] 初貝恭祐, 渡邉洋平, 岩本貢. シャカシャカに対する物理的ゼロ知識証明. 2025 年暗号と情報セキュリティシンポジウム（SCIS2025) 4D1-3, pp. 1–8, 2025.

[18] 藤原愛斗, 池田昇太, 品川和雅. テントアンドツリーに対する物理的ゼロ知識証明プロトコル. コンピュータセキュリティシンポジウム 2025 論文集, 2025.