# CS292G Research Report
# Efficient classical protocols for Quantum Verification

Riley Paddock, Grecia Castelazo, Shivane Dadi

Spring 2025

## Contents

**1 Introduction** **2**

**2 Certified Quantum Randomness** **3**
2.1 Definitions on Certified Quantum Randomness . . . 3
2.1.1 Certified Randomness as a Proof of Quantumness . . . 3
2.1.2 Protocols for Certified Randomness . . . 3
2.1.3 Cross-Entropy Benchmarking (XEB) . . . 3
2.1.4 Device-Independent Bell Inequalities . . . 4

**3 Instantaneous Quantum Polynomial Circuits** **6**
3.1 Shepherd and Bremner's IQP Protocol . . . 6
3.2 Kahanamoku-Meyer's Secret Extraction Attack . . . 7
3.3 The IQP Stabilizer Scheme and Improved Security . . . 8
3.4 Secret Extraction Attacks on the IQP Stabilizer Scheme . . . 10
3.5 Fixing Secret Extraction Attacks . . . 11

**4 Innovations on classically verifiable protocols** **12**
4.1 TCF-based quantum advantage protocols . . . 13
4.1.1 Interactive quantum advantage protocol . . . 13
4.1.2 Blueprint for implementation of $x^2 \mod N$ in Rydberg atoms . . . 14

**5 Discussion and Conclusions** **15**

# 1 Introduction

As quantum computing technology matures, demonstrating an unambiguous quantum computational advantage–where a quantum device solves a problem fundamentally faster than any classical computer–remains a paramount objective in the field. However, beyond simply achieving such an advantage, a critical challenge lies in our ability to efficiently and rigorously verify the quantum nature of the computation and the correctness of its outcome. For many tasks where quantum devices are expected to excel, such as sampling from complex probability distributions, the very act of classical verification can be as computationally demanding as simulating the quantum process itself, often requiring resources that scale exponentially with the system size. This difficulty poses a significant hurdle to convincingly establish quantum capabilities.

Addressing this verification hurdle often involves protocols designed to provide a "Proof of Quantumness."Such frameworks aim to demonstrate that a device genuinely harnesses quantum mechanical principles, rather than merely mimicking them classically. Such protocols often involve a challenge posed to the quantum device, a response (the proof) generated by the device, and a verification step – the classification of whether the correctness of the quantum computation is efficiently verifiable by a classical computer. The difficulty often lies in designing protocols where this classical verification is efficient.

One particularly vital application of these verifiable quantum processes is in the generation of truly random numbers, a cornerstone for fields like cryptography, scientific simulation, and secure communication. While classical algorithms can produce pseudo-random sequences, their inherent determinism makes them vulnerable if the underlying algorithm or seed is compromised. Quantum mechanics, with its intrinsic indeterminacy, offers a path to generating numbers that are not just unpredictable but fundamentally random. This report delves into the domain of certified quantum randomness, exploring methods to validate the quantum nature of a device and the randomness it produces.

This report will first lay out the concepts of certified randomness and its significance as a proof of quantum behavior. We will then examine protocols for achieving such certification, with a particular focus on Cross-Entropy Benchmarking (XEB), a widely used technique for validating the performance of quantum processors. We will discuss how XEB provides a statistical measure of a quantum processor's fidelity by comparing the probability distribution of its outputs for random quantum circuits to the theoretical expected distribution.

In Section 3, we present the background on Instantaneous Quantum Polynomial Circuits, as an early candidate for efficient quantum verification. These circuits were especially promising since they were conjectured to be (and still are to this day) hard to "spoof". So an adversary cannot simulate an actual IQP circuit any better than they can simulate a general quantum computer. Then, it was shown that you can embed a "secret"into these circuits such that the verifier can efficiently verify that bitstrings come from the challenge circuit if they have the secret.

However, the security of IQP circuits has been through a turbulent history of vulnerabilities and successive revisions. Mainly relating to the security of the "secret"itself. Explained in more detail in section 3, the initial protocol was shown to be weak in the sense that given the "encoding"of a circuit as a matrix, an adversary could efficiently extract the secret. So the protocol was revised to better hide this secret from the attack that exploited the initial protocol. However, this revised version was weak to another attack, the Radical Attack, which exploited how the matricies which encoded IQP circuits were distinct from random matricies. The same work that came up with the Radical Attack also proposed 3 other attacks. In response to these attacks, IQP was again revised to be secure. This most recent revision is secure as of today. However, these "secure"instances of IQP require a high number of qubits and a high circuit depth. While it is still more feasible than a Shor's algorithm based protocol, it is also still beyond near-term device capabilities.

The latest results on attacks against the recent fixes to the vulnerabilities of IQP circuits opened the problem of finding new efficient and reliable verification protocols for sampling-based proofs of quantum supremacy. In section 4, we discuss alternative protocols based on a class of cryptographic tools called trapdoor claw-free functions, and recent innovations related to postselection techniques that reduces the fidelity required to demonstrate quantum advantage. In particular, we focus on verification protocols aimed at near-term quantum devices and we discuss the blueprint of the $x^2 \mod N$ problem in a programmable Rydberg-atom-based platform from [11].

# 2 Certified Quantum Randomness

## 2.1 Definitions on Certified Quantum Randomness

Certified randomness refers to random numbers that are not only generated by a process believed to be quantum but also come with a verifiable guarantee of their quantum origin and quality. This certification ensures that the output is indeed a result of quantum mechanical principles and possesses a high degree of entropy, making it robust against classical prediction or manipulation.

### 2.1.1 Certified Randomness as a Proof of Quantumness

The certification process itself often serves as a Proof of Quantumness. If a device can consistently produce outputs that pass stringent statistical tests designed to be hard for classical systems to spoof, it provides evidence that the device is operating in a quantum regime. The core idea is that genuine quantum randomness, derived from measurements on quantum states, exhibits statistical properties that are classically intractable to reproduce efficiently for certain tasks. Thus, by verifying the randomness, one implicitly verifies the "quantumness"of the source. This is crucial for building trust in quantum devices, especially in security-critical applications. The challenge, as highlighted in the introduction, is to design verification protocols that are themselves classically efficient, avoiding the need to simulate the quantum device fully, which would defeat the purpose of quantum advantage.

### 2.1.2 Protocols for Certified Randomness

Protocols for certified randomness aim to establish the quantum nature of a device and the unpredictability of its outputs. These protocols generally involve the following components:

1. **Challenge Generation:** A classical verifier sends challenges (e.g., specifications of quantum circuits or measurement settings) to the quantum prover (the device attempting to generate randomness).
2. **Quantum Processing and Measurement:** The quantum prover executes the challenge on its hardware and performs measurements, yielding output bitstrings.
3. **Verification:** The classical verifier analyzes the output bitstrings, often comparing their statistical properties against theoretical predictions for an ideal quantum device. This step must be classically efficient.

A key distinction in protocols lies in the assumptions made about the quantum device. Some protocols operate in the device-independent regime, making minimal assumptions, while others, like those often benchmarked by XEB, are device-dependent, assuming some characterization of the device's components but testing their collective quantum behavior.

### 2.1.3 Cross-Entropy Benchmarking (XEB)

Cross-Entropy Benchmarking (XEB) is a prominent protocol used to verify the quantum state preparation and measurement capabilities of a quantum processor, and by extension, can contribute to the certification of randomness derived from such a processor. It is particularly relevant for near-term quantum devices.

The core idea of XEB is to run a set of random quantum circuits on the quantum device and compare the observed output distribution with the one predicted by theory.

**Definition 2.1 (Cross-Entropy Benchmarking Fidelity)** *Let $U$ be a random quantum circuit drawn from a specific ensemble. Let $P_{ideal}(x|U)$ be the ideal probability of measuring the output bitstring $x$ after applying $U$ to an initial state (typically $|0\rangle^{\otimes n}$). Let $P_{exp}(x|U)$ be the experimentally observed probability of measuring $x$ from the quantum device. The **linear cross-entropy fidelity** for a single circuit $U$ is estimated by sampling $N_s$ bitstrings $\{x_i\}$ from the device and calculating:*

$$F_{XEB}(U) = 2^n \left( \frac{1}{N_s} \sum_{i=1}^{N_s} P_{ideal}(x_i|U) \right) - 1$$

*The factor $2^n$ accounts for the normalization if $P_{ideal}$ were a uniform distribution. If the device is perfect and produces samples according to $P_{ideal}$, then $\mathbb{E}[P_{ideal}(x_i|U)] = \sum_x P_{ideal}(x|U)^2$. For chaotic quantum circuits, this sum approaches $2 \cdot 2^{-n}$ (the Porter-Thomas distribution), leading to an expected $F_{XEB} \approx 1$. If the device outputs a uniform distribution (e.g., due to decoherence), then $P_{ideal}(x_i|U)$ would average to $2^{-n}$, resulting in $F_{XEB} \approx 0$.*

The overall XEB fidelity is then averaged over many different random circuits $U$. A consistently high XEB fidelity (close to 1) indicates that the quantum device is faithfully executing the quantum circuits and producing outputs that are strongly correlated with the ideal quantum evolution. This provides strong evidence against classical spoofing, as efficiently calculating $P_{\text{ideal}}(x|U)$ for random quantum circuits is believed to be classically hard

Verification Process with XEB:

1. **Challenge:** The verifier (classical computer) generates a set of random quantum circuits. These circuits are typically chosen from a family that is hard to simulate classically but for which the output probabilities $P_{\text{ideal}}(x|U)$ can be computed (albeit expensively, this is done by the verifier who has significant classical resources, or for small enough systems for direct verification).

2. **Proof:** The quantum device executes these circuits and returns a collection of output bitstrings for each circuit.

3. **Verification:** The verifier computes the $F_{XEB}$ score. If the score is sufficiently high, it suggests that the device is indeed performing quantum computations as described by the circuits. The randomness generated during these experiments (the output bitstrings themselves) can then be considered certified to the extent that the XEB fidelity validates the quantum nature of the process.

While XEB is a powerful tool for benchmarking quantum processors and serves as evidence of "quantumness,"the direct extraction of certified random bits from XEB experiments for cryptographic use requires further considerations, such as protocols for randomness expansion based on the verified quantum behavior. However, a high XEB score is a strong indicator that the device is not producing outputs that a classical algorithm could easily predict or replicate for the given circuits.

### 2.1.4 Device-Independent Bell Inequalities

Another category for quantum verification is offered by device-independent protocols, which allow for the certification of quantum properties without needing to trust the internal workings or detailed specifications of the quantum devices used. The foundation for many such protocols lies in Bell's theorem [4] and its experimental tests using Bell inequalities. One of the most crucial and widely recognized of these is the Clauser-Horne-Shimony-Holt (CHSH) inequality [8].

The CHSH framework considers a scenario with two separated parties, commonly referred to as Alice (A) and Bob (B). Each party receives an input (Alice receives $x$, Bob receives $y$) and produces an output (Alice produces $a$, Bob produces $b$). In the simplest case, the inputs are binary, $x, y \in \{0, 1\}$, and the outputs are also binary, $a, b \in \{+1, -1\}$. The experiment involves running many rounds, where in each round Alice and Bob randomly choose their inputs, perform measurements on their respective parts of a shared physical system (e.g., entangled quantum particles), and record their outputs.

The CHSH inequality is formulated in terms of the expectation values of the products of Alice's and Bob's outcomes, $E(x, y) = \langle a_x b_y \rangle$, where $a_x$ is Alice's outcome given input $x$, and $b_y$ is Bob's outcome given input $y$. The inequality states that for any theory based on local hidden variables (i.e., any classical theory where properties are predetermined and local interactions are respected), the following combination of these correlations must hold:

$$S = |E(0, 0) + E(0, 1) + E(1, 0) - E(1, 1)| \leq 2$$

This value, $S$, is often called the CHSH value. Classical physics, or any local realistic theory, dictates that $S$ cannot exceed 2.

However, quantum mechanics predicts that if Alice and Bob share an entangled state (e.g., a maximally entangled pair of qubits) and choose their measurement settings appropriately, they can achieve a value of $S$ that violates this bound. Specifically, quantum mechanics allows for:

$$S_{\text{quantum}} \leq 2\sqrt{2} \approx 2.828$$

This is known as Tsirelson's bound.

The experimental observation of a CHSH value $S > 2$ (a "Bell violation") provides strong evidence against local realism and serves as a device-independent proof that the devices are operating in a genuinely quantum manner, specifically that they are exploiting entanglement. The "device-independent"nature arises because the conclusion is drawn solely from the observed input-output statistics $(a, b|x, y)$, without making any assumptions about the type of quantum state shared or the specific quantum measurements being performed. The only assumptions are that Alice and Bob can make independent random choices for their inputs and that their devices are causally separated during the measurement process (to prevent classical communication from explaining the correlations).

Thus, a Bell test based on the CHSH inequality acts as a verification protocol:

1. **Challenge:** The verifier (or the experimental setup itself) provides random inputs $x$ to Alice's device and $y$ to Bob's device over many trials.

2. **Quantum Processing and Measurement:** Alice's and Bob's (potentially untrusted) quantum devices produce outputs $a$ and $b$, respectively.

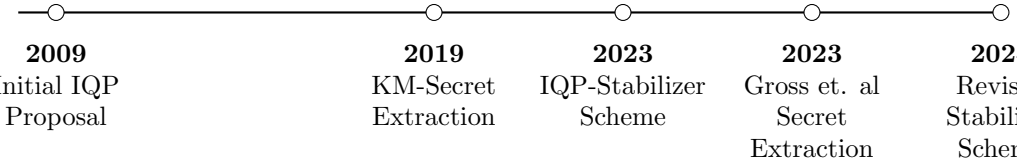
3. **Verification:** The verifier collects the statistics of $(a, b|x, y)$ from many runs, computes the CHSH value $S$. If $S > 2$ (and ideally approaches $2\sqrt{2}$ for high-quality entanglement and measurements), it certifies the quantum nature (non-locality and entanglement) of the shared system and the operations, regardless of the internal details of the devices.

The CHSH inequality and its variations are fundamental tools in quantum information science, underpinning protocols for device-independent quantum key distribution, certified randomness generation, and benchmarking quantum devices.

# 3 Instantaneous Quantum Polynomial Circuits

Instantaneous Quantum Polynomial(IQP) Circuits were proposed by Shepherd and Bremmer in 2009 [13] as one of the first candidates for a classically verifiable proof of quantum advantage. Simply, they are a special class of quantum circuits which only consist solely of $R_X$ gates with some fixed angle. As we will explore later, there is a way to encode these circuits such that you can efficiently verify that bitstrings were sampled from the measured outputs of these circuits.

However, the development of IQP circuits has been a back-and-forth between multiple authors breaking the security of the protocol and then fixing and then breaking again. Here is a timeline of the IQP developments up to this year:




Outlining this timeline in more detail; The initial paper by Shepherd and Bremmer conjectured that their protocol was secure against classical adversaries trying to simulate the IQP circuit. They posted a "challenge"circuit of 487 qubits with a reward of \$25. Then, in 2019, an attack (KM-Attack) [10] which worked by extracting a "secret"string necessary to the protocol security showed that it was insecure (And won the \$25!). Then in August 2023 the IQP-Stabilizer Scheme (Bremmer et al.)[6] was posted which fixed this vulnerability. It also posed a new challenge circuit on 300 qubits which was proposed to be secure. By December 2023 a paper was posted by Gross and Hangleiter[9] which beat this 300-qubit challenge using both upgraded versions of the KM-Attack and new attacks. In response to this, in December 2024, Bremmer posted a revision to the IQP stabilizer scheme secure against the new attacks. Gross and Hangleiter shared, in a revison to their paper (Feb 2025), that the revisions were secure against all approaches they tried.

## 3.1 Shepherd and Bremner's IQP Protocol

Here is a more in depth outline of the Initial IQP protocol. Note that we adopt some of the notation of the later IQP Stabilizer Scheme so that this aligns with later sections. But as a result it doesn't line up exactly with the initial paper itself. As briefly stated above, an IQP circuit is a circuit of all $R_X$ gates of some fixed angle $\theta$.

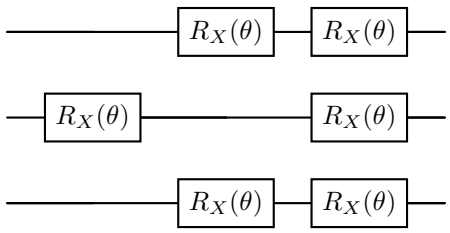



Figure 1: Example of an IQP circuit on 3 qubits

Since the $X$ rotation gate can be expressed as $R_X(\theta) = e^{i\frac{\theta}{2}X}$ An IQP circuit has realized a unitary of the following form:

$$U = exp(i\frac{\theta}{2}[I \otimes X \otimes I + X \otimes U \otimes X + X \otimes X \otimes X])$$

Note, thus is an example to match Figure 1. Importantly, this structure gives rise to a nice matrix representation. The matrix is $m \times n$ where $m$ is the depth of the circuit and $n$ is the number of qubits. So for Figure 1 the matrix would be:

$$H = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

**Key Theorem of Shepherd and Bremmer**

This matrix $H$ we defined above, serves not only as a descriptor of our IQP circuit but also a generator matrix for a code. In particular, the set of codewords generated by $H$ is:

$$\mathcal{C} = \{Hd^T | d \in \{0,1\}^r\}$$

Importantly, Shepard and Bremmer derived the following results for IQP circuits.

**Theorem 1** *Let $X$ be a random variable representing the bitstring outputs of an IQP circuit given by code-generator matrix $H$. Then for any bitstring $s$:*

$$P(Xs^T = 0) = E_{c\sim\mathcal{C}}[cos^2\left(\theta(n_s - 2 \cdot wt(c)\right)]$$

*Where $C$ is the code generated by $P$, $\theta$ is the rotation angle of our $R_X$ gates, $wt(c)$ is the Hamming weight of a codeword, and $n_s$ is the rows of $P$ not orthogonal to $s$.*

This is important because this expectation is efficiently computable! The ability to know this expectation, if you know the string $s$, is how our verification can be classically efficient. We will now use this fact to construct a proof of quantumness.

#### The Initial IQP Protocol

The main idea we want to use here is that we choose $H$ and $s$ and pre-compute the expectation from Theorem 1 and then send the circuit. Then, because the "proof"we recieve is many bitstrings, We can hypothesis test if the responses (which should be modeled by the random variable $X$) satisfy Theorem 1, namely if $P(Xs^T = 0)$ matches our expectation.

To appropriately set up the code in a way that is later vunerable, we need to mention two types of codes; the "Quadratic-Residue Code"and the "Doubly Even Code". A "Doubly Even Code"is simply a code where every codeword has hamming weight divisible by 4. A Quadratic Residue Code is beyond the scope of this paper but we will note a couple key facts:

- Length of the code, $q$, is prime and $q + 1 \equiv 0 \mod 8$
- Code is cyclic, generated by a single codeword, has a 1 at each quadratic residue mod $q$, and 0 elsewhere.
- It is a doubly even code
- With secret $s$ circuit matrix $H$, if $Hs = \vec{1}$, then $P(Xs^T = 0) = 0.854$

This last fact is why this code is desireable, it gives a large bias in the direction of $s$. So Alice starts with a quadratic residue code $\mathcal{C}$. Then, she computes a generator matrix $H$ and a secret vector $s$ such that $Hs^T = \vec{1}$.

### Obfuscation of the secret $s$

Now, to make sure the adversary can't simply solve the linear system $Hx = \vec{1}$ to find $s$. We want to "hide"the secret $s$. We do this by adding some rows to $H$ which are orthogonal to $s$. This preserves the code and thus the expectation. Then we permute the rows and generate the challenge circuit from this altered $H'$. For future reference, we denote $H_s$ as the obfuscated $H'$ restricted to the rows which are not-orthogonal to $s$.

### Final Notes on initial IQP

As outlined above the responses can be verified by testing if they have the expected "bias"towards $s$. This was doubly promising not only because it was efficiently verifiable, but it was also probabilistic, so it could account for NISQ levels of noise. But the security of this protocol was based on *conjectured* hardness of simulating IQP circuits (which still holds) and the hardness extracting the secret $s$ from our altered $H'$. This ends up being insecure. The authors published a $25 challenge with their paper in 2010 that used a particular Quadratic-Residue Code.

## 3.2 Kahanamoku-Meyer's Secret Extraction Attack

In 2023, Gregory Kahanamoku-Meyer published work exploiting a vunerability in the IQP protocol. In particular, it found a way to extract the secret $s$ from $P'$ with probability $\frac{1}{2}$. Clearly, once the secret $s$ is known, a classical prover can efficiently generate classical samples that perfectly mimic the required correlation with $s$, thus passing the verifier's test without needing a quantum computer or simulating one.

**Algorithm 1** Kahanamoku-Meyer's Secret Extraction Attack

Let $m$ be the vector obtained by adding all rows of $H'$.
Pick $d \xleftarrow{\$} \{0,1\}^n$. Keep this fixed
**for** $i \leftarrow 1$ **to** $2n$ **do**
Generate $\{m_i\}$ as

$$e \xleftarrow{\$} \{0,1\}^n$$

$$m_i = m + \sum_{h \cdot e = h \cdot d = 1} h$$

Form a matrix $M$ with $m_i$ as its rows
Solve for all vectors $s_i$ such that $Ms_i = \vec{1}$ ▷ Each $s_i$ serves as a candidate secret
**if** $H_{s_i}$ is a doubly even code **then**
**return** $s_i$
**end if**
**end for**

**Correctness Sketch**

This algorithm exploits the Quadratic-Residue Code more than any of the other structure of IQP. The first key observation is that for the true secret $s$:

$$m \cdot s = 1$$

This follows from $m \cdot s = \sum(h \cdot s)$ and either $h$ was added for obfuscation and $h \cdot s = 0$ or, $h$ is from the original encoding $H$, so $h \cdot s = 1$. The original encoding had $q$ rows, which is prime, thus odd. So an odd number of these inner products are non-zero. So $\sum(h \cdot s) = 1$

Then, considering the second half of each $m_i$:

$$\left( \sum_{p \cdot e = p \cdot d = 1} p \right) \cdot s = 0$$

This is simply 1 or 0 with 50/50 odds by a parity argument. Then, it can be shown that if a string $s$ both satisfies $Ms = 1$ and the sub-matrix $H_s$ of vectors not orthogonal to $s$ generates a Doubly-Even Code. Then $s$ must be the true secret $s$. Then, by the above argument, we get $Ms = 1$ with probability $1/2$

**Performance**

This algorithm was shown to run in $O(n^3)$ and performed with $O(n^2)$ scaling by a parallelized vector computation subroutine on the processor used. With this algorithm Kahanamoku-Meyer solved the $25 challenge, conjectured to need a $\sim 400$ qubit computer to efficiently prove, in under a second.

## 3.3 The IQP Stabilizer Scheme and Improved Security

Following, what they called the KM-Attack, Bremmer worked with Cheng and Ji to upgrade the IQP protocol. They called this upgraded method the *IQP Stabilizer Scheme*. A key security assumption that they based this upgraded version on is their conjectured hardness of the *Hidden Structured Code*(HSC). To adopt more of their notation, let $U_{H,\theta}$ be the $IQP$ circuit made with matrix $H$ and angle $\theta$. Then define the *correlation function* as:

$$\langle \mathcal{Z}_s \rangle = \langle 0^n | \, U_{H_s, 2\theta} \, |0^n\rangle$$

Recall, $H_s$ is the matrix $H$ restricted to the rows such that $h \cdot s = 1$. This is related to the $P(Xs^T = 0)$ from the initial protocol by:

$$P(Xs^T = 0) = \frac{1}{2}(\langle \mathcal{Z}_s \rangle + 1)$$

This new correlation function is what we use to verify.

**Hidden Structure Code(HSC)**

First, we define a family of matrix-vector pairs which are useful for IQP. Let $H \in \{0,1\}^{m\times n}$ and $s \in \{0,1\}^n$. Let $\mathcal{H}_{n,m,g}$ = {(H,s)} be the family of pairs $H, s$ such that:

$$rank(H_s^T H_s) = g \quad rank(H) = n$$

$$|\langle \mathcal{Z}_s \rangle|^2 = 2^{-g/2}$$

Then, the key idea of the HSC is the following conjecture:

**Definition 3.1** *The Hidden Structure Code Problem*

*There is an algorithm which efficiently samples from $(H, s) \leftarrow \mathcal{H}_{n,m,g}$ for certain values of $n, m, g$ such that no classical poly-time adversary can find $s$ given $H$ with high probability*

The power of this, is that we can find which parameters $n, m, g$ we consider secure, and then just from those parameters we can sample secure IQP instances $H, s$. In particular, and important to the future attacks on the stabilizer scheme, these matricies $H$ have the form seen in Figure 2

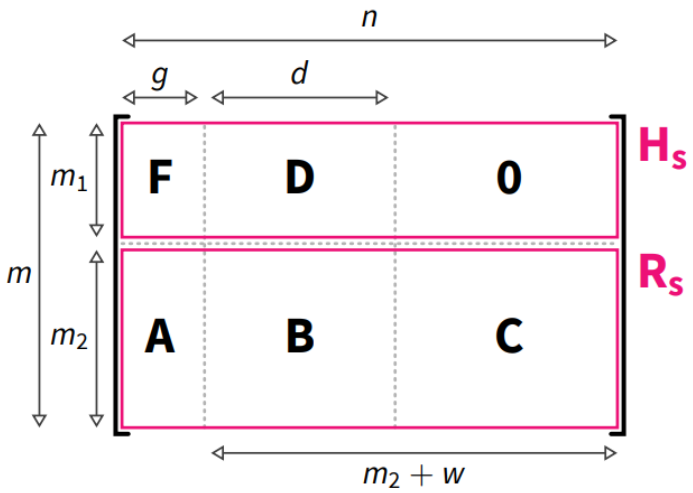



Figure 2: Breakdown of the generator matricies used in the IQP Stabilizer Scheme

In particular, the blocks of $H$ satisfy certain constraints:
1) $D$ is full rank and gives a doubly-even code such that $D^T D = 0$
2) $F$ is full rank and $D^T F = 0$
3) $C$ is full rank (Since $F, D, C$ are full rank and orthogonal, $H$ is full rank)

**A note on parameters:** Recall here that $n$ is the number of qubits required, and $m$ is the depth of the circuit. Further, $g$ is the "log of the power of the statistical test". Finally, you may notice that the breakdown in Figure 2 has two extra free parameters, $d$ and $m_1$. These get sampled from a distribution which is a function of $m, g$ and the mean values are: $m_1 \sim \frac{3}{20}(m-g) \quad d \sim \frac{3}{160}(3m-23g) = \frac{3}{8}(m_1 - g)$

**Security Versus KM-Attack**

It is shown that for $(H, s) \leftarrow \mathcal{H}_{n,m,g}$ and random vector $d$ then:

$$P[s \in ker(H_d^T H_d)] = 2^{-g}$$

This probability is the probability that:

$$\sum_{h\cdot d = h\cdot e = 1} h \equiv 0$$

The power of the KM attack was that this had probability 1/2, which was a result of the Quadratic Residue Code having $g = 1$. But, the authors show that they can generate instances where $2^{-g/2} = \Omega(1/poly(n))$.

### 3.4 Secret Extraction Attacks on the IQP Stabilizer Scheme

As outlined in the timeline, soon after the IQP-Stabilizer Scheme was posted, it was broken. Gross and Hanglieter gave new secret extraction attacks which found the secret key from new challenge instances. But, in response, the IQP Stabilizer Scheme was updated. So to make the most sense of analyzing it's security we first need to describe the 3 new attacks from Gross and Hangleiter.

#### Radical Attack

This attack exploits the fact that $H^T H$ will have a much larger kernel than a random matrix.

**Algorithm 2** Radical Attack

Let $G = H^T H$
Solve for $K$ a column generating matrix for $ker(G)$
$S \leftarrow$ the support of columns of $HK$
Solve for $s$ in the linear system $Hs = \vec{1}_S$
**return** $s$

This attack works because $range(H(ker(G))) \subseteq range(H_s)$ and with high probability, this subspace is very large. Then since $s$ is in this subspace, we have with high probability that we can solve for it. In particular, Gross and Hanglieter show analytically and empirically that when $n - g - m_2 > 7$, the radical attack finds the secret with probability $1 - negl$.

#### Lazy Linearity

This attack is an upgrade to the KM-attack with 3 new parameters. the ambition $A$, the endurance $E$, and the significance threshold $g_{th}$. The algorithm is as follows:

**Algorithm 3** Lazy-Linearity

**Let** i = 0
**while** $i < E$ **do**
  $d \xleftarrow{\$} \{0,1\}^n$
  $G_d = H_d^T H_d$
  **if** $dim(ker(G_d)) < A$ **then**
    **for** $x \in ker(G_d)$ **do**
      **if** $H_x$ is doubly even and $rank(H_x^T H_x) \leq g_{th}$ **then**
        **return** $x$
      **end if**
    **end for**
    Delete $h'$
  **end if**
  $i \leftarrow i + 1$
**end while**

The Lazy Linearity attack works very similarly to the KM attack. But where the KM-attack found a specific approach to get a small $Ker(G_d)$, this attack instead iteravely tries $E$ times and only tries to find the secret if $ker(G_d)$ is "small enough"to explore. Gross and Hanglieter note that the combination of of Radical and Lazy-Linear Attacks seemingly fully solve Quadratic Residue Codes, since their performances are complimentary on the number of qubits. (Figure 3 )

#### Security against Hammings Razor

Hamming's Razor is very different than the Linearity Attack. Rather than try to expose the secret directly. It simply shaves off columns which have been added to obfuscate the secret. Making it easier to perform the Radical and Lazy Linearity attacks. It works off a simple idea:

If $Hd = e_i$ where $e_i$ is the $i$'th standard basis vector, then the $i$'th row of $H$ is orthogonal to $s$ and we can remove it. If we reference Figure 2, the matrix $H$ is broken into the secret-encoding rows $(F|D|0)$ and the obfuscating rows $(A|B|C)$. All this theorem requires is that $(F|D|0)d \neq e_k$.

We know that $e_k$ is not in $range(D)$ since $D$ generates a doubly even code so $w_{ham}(Dd) \equiv 0 \mod 4$, and $w_{ham}(e_k) = 1$. Since $D$ is full rank, it follows that $range((F|D|0)) = range(D)$. Therefore, $e_k \notin range(F|D|0)$.

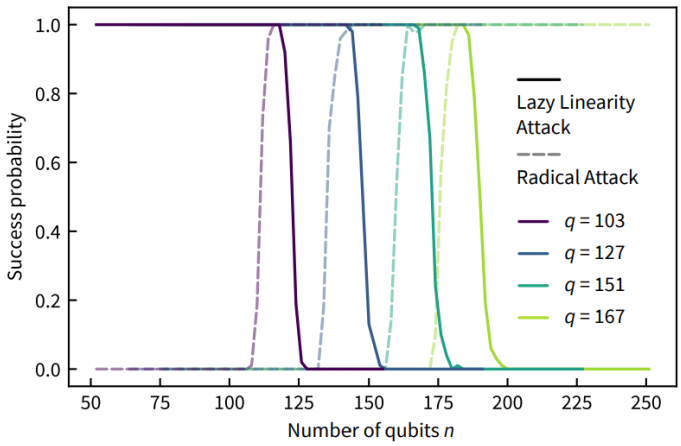



Figure 3: Performance of the Radical and Lazy Linear attacks on Quadratic Residue Code based IQP cirquits with varying number of qubits

Gross and Hanglieter further this attack by realizing that not only are standard basis vectors not in the image, but low hamming weight vectors have low probability of being in the image. With finely tuned probability metrics and a clever algorithm (As they outline in their Lemma 6 and Algorithm 4) one can actually remove every obfuscation row of $H$ and easily solve for $s$.

## 3.5 Fixing Secret Extraction Attacks

### The Radical Attack

The way to circumvent the Radical attack is actually quite simple. We want to make $ker(H^T H)$ smaller, since it's size is what lets it success with high probability. To do this, we again reference Figure 2 and enforce that $(B|C)$ is full rank. This is possible if we force $m_2$ to satisfy $m_2 \geq n-g$ (Recall that the Radical attack succeeded when $n-g-m_2 > 7$) when initially we only guaranteed $m_2 \geq n-g-d$.

### Lazy Linearity

The key bottleneck on the Lazy Linearity, is the expected number of iterations $(E)$ needed to find a random vector $d$ such that $Ker(G_d)$ is small $(\leq A)$. In particular, if you keep $m$ relatively small (They suggest $\leq 1200$) but KEEP $n-(m/2)$ larger than any feasible $A$, the number of iterations becomes exponential.

### Hamming's Razor

Surprisingly, the authors show that the "safe"parameters for the Radical Attack and the Linearity Attack are actually more susceptible to Hamming's Razor than their initial protocol. However, they identify that if their matrix $H$ is sparser, then it is more resilient to Hamming's Razor. They make it sparser in two ways:

1. Construct $D$ in the following way. Find $k$ full rank doubly-even code generating matricies $D_i$ of size $m_0 \times d_0$ where $m_0 \cdot k = m_1$. Let $E = diag(D_1, D_2, \ldots, D_k) \in \{0,1\}^{km_0, kd_0}$ and let $K$ be a random full rank matrix in $\{0,1\}^{d_0 k \times d}$. Then define $D = EK \in \{0,1\}^{m_1 \times d}$
2. Set $(A|B)$ be a random sparse matrix and then alter the first column of $A$ to that $(A|B)s = 0$.

### Secure IQP instance

With the combination of all of these approaches, they constructed an instance which was secure to all attacks. However, this attack had parameters $n = 700, m = 1200$. Which means a quantum computer would need 700 qubits running a circuit of depth 1200 to sample outputs. This is beyond NISQ capabilities, so not a feasible candidate for current quantum verification protocols.

# 4 Innovations on classically verifiable protocols

The latest results on attacks [9] against the newest fixes to the vulnerabilities of IQP [6] using obfuscated circuits opened the problem of finding new efficient and reliable verification protocols for sampling-based proofs of quantum supremacy. Some recent protocols rely on a post-quantum secure trapdoor claw-free (TCF) family of function pairs $f_0, f_1 \to \{0,1\}^n \to \{0,1\}^m$.

The quantum prover can successfully answer either challenge in the qubit certification protocol by measuring the state in in the standard or Hadamard basis. By contrast, no classical algorithm can succeed at this task.

However, the TCFs have a challenging requirement: the adaptive hardcore bit property. A recent promising candidate, which is based on a connection to Bell's inequality, bypasses the need for the adaptive hardcore bit property while maintaining essentially the same quantum circuit complexity and assumptions.

We begin by comparing different cryptographic assumptions on interactive quantum advantage protocols. All of the TCFs listed in Table 1 can be used within the interactive protocol, each offering distinct advantages. The TCF based on the Diffie-Hellman method demonstrates quantum advantage at key sizes of 160 bits; however, it requires the quantum implementation of Euclid's algorithm. For this reason, Kahanamoku-Meyer et al. focused on designing quantum circuits that implement Rabin's function.

| Problem | Trapdoor | Claw-free | Adaptive hardcore bit | Asymptotic complexity |
|---|---|---|---|---|
| LWE | ✓ | ✓ | ✓ | $n^2 \log^2 n$ |
| $x^2 \mod N$ | ✓ | ✓ | ✗ | $n \log^2 n$ |
| Ring-LWE | ✓ | ✓ | ✗ | $n \log^2 n$ |
| Diffie-Hellman | ✓ | ✓ | ✗ | $n^3 \log^2 n$ |
| Shor's algorithm | – | – | – | $n^2 \log n$ |

Table 1: Cryptographic constructions for interactive quantum advantage protocols. For $n$ number of bits in the function's input string.

### Quantum certification protocol

**TCF functions** A TCF pair $f_0, f_1 \to \{0,1\}^n \to \{0,1\}^m$ is a pair of injective functions, with the same image satisfying the following property: *With* knowledge of a secret trapdoor it is possible to efficiently (classically) compute the two preimages $x_0$ and $x_1$ given $y(f_0(x_0) = y)$, *but without* the trapdoor there is no efficient quantum algorithm that can recover such a triple: $(x_0, x_1, y)$, known as a *claw* for any $y$. The key idea is that while a quantum device cannot compute a *claw*, it can hold an image $y$ and a superposition

$$\frac{1}{\sqrt{2}} \left( |0\rangle |x_0\rangle + |1\rangle |x_1\rangle \right)$$

over the two preimages of $y$, by evaluating $f$ on a uniform superposition of inputs. To exploit this superposition, perform a Fourier (Hadamard) basis measurement on all except the first qubit of the state yielding a string $d \in \{0,1\}^n$ and the device will hold for $c = d \cdot (x_0 \oplus x_1)$, in

$$\frac{1}{\sqrt{2}} \left( |0\rangle + (-1)^c |1\rangle \right)$$

A Fourier measurement of the single-qubit state will yield the bit $c$. This one Fourier measurement is the differentiator between quantum and classical devices. The output $(d, c)$ is inherently difficult to reproduce in a classical setting because it depends on both elements, $x_0$ and $x_1$, being in superposition. Moreover, the claw-free property indicates that it is computationally challenging to simultaneously hold both $x_0$ and $x_1$. The aforementioned steps suggest the following quantum certification protocol. The ground work on cryptographic tests of 'quantumness' [5] defines the quantum certification protocol.

**Algorithm 4** Quantum certification protocol

1: The (classical) Verifier generates a TCF pair, along with a trapdoor.
2: Send the function pair to the (quantum) Prover.
3: The (quantum) Prover returns an image $y$ on the TCF pair.
4: The (classical) Verifier challenges the (quantum) Prover by randomly asking for a preimage of $y$ of a bit $c$ and an $n$-bit string $d$ such that

$$d \cdot (x_0 \oplus x_1) = c$$

5: The (quantum) Prover measures in $Z$- or $X$-basis to return an answer.
6: The (classical) Verifier checks the validity by using the trapdoor to compute the two preimages $x_0, x_1$ of $y$.

**Circumventing the adaptive hardcore bit**

Some of these alternate protocols rely on a trapdoor claw-free functions, which we define below. The TCFs were first used in quantum cryptographic tasks (1) to give classical homomorphic quantum circuit encryption and (2) to generate cryptographic certifiable quantum randomness from an untrusted blackbox device.

**Definition 4.1 (Trapdoor Claw-free Functions)** *Let $\lambda$ be a security parameter, $I$ a set of function indices, and $X_i$ and $Y_i$ finite sets for each $i \in I$. A family of functions*

$$\mathcal{F} = \{f_i : X_i \to Y_i\}_{i \in I}$$

*is called a trapdoor claw-free (TCF) family if the following conditions are met:*

1. *Efficient Function Generation. There exists an efficient probabilistic algorithm* Gen *generating a key and associated trapdoor data $(i, t_i)$ such that,*

$$(i, t_i) \leftarrow Gen(1^\lambda)$$

2. *Trapdoor Injective Pair*

3. *Claw-free. For any non-uniform probabilistic polynomial time (nu-PPT) classical Turing machine $\mathcal{A}$, there exists a negligible function $\epsilon(\cdot)$ such that*

$$\mathbb{P}[f_i(x_0) = f_i(x_1) \wedge x_0 \neq x_1 | (x_0, x_1) \leftarrow \mathcal{A}(i)] < \epsilon(\lambda).$$

4. *Efficient Superposition. There exists an efficient quantum circuit that on input key $i$ prepares the state*

$$\frac{1}{\sqrt{|X_i|}} \sum_{x \in X_i} |x\rangle |f_i(x)\rangle$$

## 4.1 TCF-based quantum advantage protocols

The desirable properties of TCF constructions for quantum verification are small key size and efficient quantum circuits. The second construction is inspired by the Rabin cryptosystem, using the function $f_N(x) = x^2 \mod N$, where $N$ is the product of two primes. The two TCF-based constructions were based on Rabin's function and the Diffie-Hellman problem.

### 4.1.1 Interactive quantum advantage protocol

The interactive quantum advantage protocol proof is based on the random oracle model (ROM)– a computational model in which both the quantum prover and the classical verifier can query a third-party oracle that returns a random but consistent output for each input.

**Algorithm 5** Interactive quantum advantage protocol

**First round.** The (classical) verifier selects a specific function from a trapdoor claw-free family and the (quantum) prover evaluates it on a superposition of qubits, giving rise to a multi-qubit superposition over two classically hard-to-compute bitstrings.
**Second round.** Condense the information contained in the prover's superposition state onto an ancilla single-qubit state.
**Third round.** Perform a CHSH-type measurement, resulting in a cryptographically protected outcome correlated with the secret.

**Completeness** An error-free quantum device honestly following the interactive protocol will cause a verifier to return *Accept* with $p_x = 1$ and $p_{\text{CHSH}} = \cos^2 \pi/8 \approx 0.85$.

**Soundness.** Assume the function family used in the interactive protocol is claw-free. For function family's input strings of length $n$ and a negligible function of $n$, $\epsilon$. Then $p_x$ and $p_{\text{CHSH}}$ for any classical prover satisfy the relation

$$p_x + p_{\text{CHSH}} - 4 < \epsilon(n).$$

**Connection to Bell's inequalities**

The connection with the CHSH game arises from the fact that if $p_x = 1$, the bound requires $p_{\text{CHSH}} < 3/4 + \epsilon(n)$ for a classical and $p_{\text{CHSH}} \approx 0.85$ for a quantum device, matching the classical/quantum probabilities for CHSH.

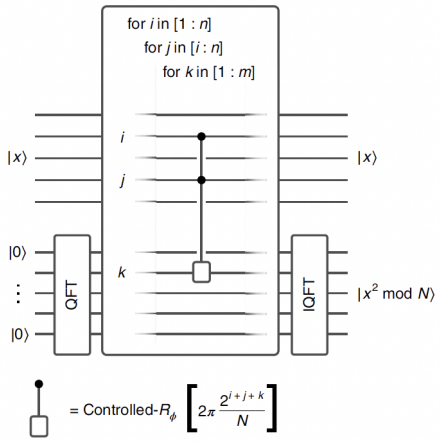



Figure 4: Phase circuit implementing $x^2 \mod N$ on a length-$n$ input register and $m = n + \mathcal{O}(1)$ output register.

### 4.1.2 Blueprint for implementation of $x^2 \mod N$ in Rydberg atoms

As a physical setting, envision a 3D-system of trapped $^{87}$Rb atoms in an optical tweezer array. Gates between atoms are facilitated by coupling to highly excited Rydberg states $|r\rangle$, enabling the Rydberg blockade mechanism. This mechanism can be explained as follows: when a single atom is driven to the Rydberg state, all other atoms within a blockade radius $R_b$ become off-resonant to the drive, resulting in suppressed excitation. In Rydberg atom systems, setting the blockade radius to be considerably larger than the atom spacing allows for multi-qubit entangling operations, precisely enabling the multi-qubit controlled-phase rotations that are necessary to build the phase 'circuits' described above.

**Goal: Applying a $C^k R_\phi^\ell$** This gate applies phase rotations $\{\phi_1, \phi_2, \ldots, \phi_\ell\}$ to target qubits $\{j_1, j_2, \ldots, j_\ell\}$ if all $k$ control qubits $\{i_1, i_2, \ldots, i_k\}$ are in the $|1\rangle$ state.

**Experimental routine**

1. Sequential resonant $\pi$-pulses applied on the $|0\rangle \leftrightarrow |r\rangle$ transition for $k$ control atoms. This creates a blockade if any of the control qubits are in the $|0\rangle$ state.

2. Off-resonant drive to the $|1\rangle \leftrightarrow |r\rangle$ transition of each target for a time duration

$$T = \frac{2\pi}{\sqrt{\Omega^2 + \Delta^2}}$$

   for detuning $\Delta$ and (driving) Rabi frequency $\Omega$. This step imprints a phase

$$\phi = \left(1 - \frac{\Delta}{\sqrt{\Delta^2 + \Omega^2}}\right)$$

   on the $|1\rangle$ state, in the absence of blockade.

3. Sequential resonant $-\pi$ pulses to return the control atoms to the original state

   Tuning the values of $\phi_i$ for each target qubit corresponds to adjusting $\Delta$ and $\Omega$ in the second step.

In summary, exploiting the Rydberg mechanism that allows for direct implementation of multi-qubit-controlled arbitrary phase rotations circumventing the need for decomposition into universal two-qubit operations; [11] outlines a verifiable test of quantum advantage that can be achieved on $\sim 10^3$ qubits and gate depth $\sim 10^5$.

# 5 Discussion and Conclusions

Our initial motivation in choosing the topic of efficient protocols for quantum verification arrose our interest in the recent demonstrations of certified randomness [12] on near-term devices (namely a trapped-ion processor). We first aimed to devise a small-scale demonstration with fewer qubits on commercially available hardware; however, the minimum required resources for this demonstration was very considerable and outside our reach in computing resources and term-project time-constraints. We were then posed with the question of "can we explore quantum verification through a more structured class of circuits such as IQP?', namely IQP –provenly hard to simulate classically. These circuits were especially promising, since they were conjectured to be hard to "spoof". So an adversary cannot simulate an actual IQP circuit any better than they can simulate a general quantum computer. Then, it was shown that you can embed a "secret" into these circuits so that the verifier can efficiently verify that the bitstrings come from the challenge circuit if they have the secret. Nevertheless, the security of IQP circuits experienced a troubled history of vulnerabilities and ongoing revisions, as detailed in section 3.

Throughout this survey, we have expanded our horizons on different angles to quantum verification by learning the aforementioned history of the IQP circuits, vulnerability breaks and revisions; and especially by learning the requirements for alternative methods in Section 4. We discussed TCF-based tests, especifically those meeting the desirable properties of TCF constructions for quantum verification such as small key size and efficient quantum circuits.

Tying back to our interest in near-term implementation, the verification method discussed in Section 4 provides an analysis of robustness and error mitigation through post-selection. Although the details of this implementation are beyond the scope of our course paper, it is a strong candidate for near-term demonstrations. An interesting future direction for this project would be to experiment with the Rydberg atom-based implementations.

# References


[1] Scott Aaronson and Sam Gunn. *On the Classical Hardness of Spoofing Linear Cross-Entropy Benchmarking*. 2020. arXiv: 1910.12085 [quant-ph]. URL: https://arxiv.org/abs/1910.12085.

[2] Omar Amer et al. *Applications of Certified Randomness*. 2025. arXiv: 2503.19759 [quant-ph]. URL: https://arxiv.org/abs/2503.19759.

[3] Frank Arute et al. "Quantum supremacy using a programmable superconducting processor". In: *Nature* 574.7779 (2019), pp. 505–510.

[4] John S Bell. "On the Einstein Podolsky Rosen paradox". In: *Physics Physique Fizika* 1.3 (1964), p. 195.

[5] Zvika Brakerski et al. *A Cryptographic Test of Quantumness and Certifiable Randomness from a Single Quantum Device*. 2021. arXiv: 1804.00640 [quant-ph]. URL: https://arxiv.org/abs/1804.00640.

[6] Michael J. Bremner, Bin Cheng, and Zhengfeng Ji. "Instantaneous Quantum Polynomial-Time Sampling and Verifiable Quantum Advantage: Stabilizer Scheme and Classical Security". In: *PRX Quantum* 6.2 (Apr. 2025). ISSN: 2691-3399. DOI: 10.1103/prxquantum.6.020315. URL: http://dx.doi.org/10.1103/PRXQuantum.6.020315.

[7] Michael J. Bremner, Ashley Montanaro, and Dan J. Shepherd. "Average-Case Complexity Versus Approximate Simulation of Commuting Quantum Computations". In: *Phys. Rev. Lett.* 117 (8 Aug. 2016), p. 080501. DOI: 10.1103/PhysRevLett.117.080501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.117.080501.

[8] John F Clauser et al. "Proposed experiment to test local hidden-variable theories". In: *Physical Review Letters* 23.15 (1969), p. 880.

[9] David Gross and Dominik Hangleiter. "Secret extraction attacks against obfuscated IQP circuits". In: *arXiv preprint arXiv:2312.10156* (2023).

[10] Gregory D. Kahanamoku-Meyer. "Forging quantum data: classically defeating an IQP-based quantum test". In: *Quantum* 7 (Sept. 2023), p. 1107. ISSN: 2521-327X. DOI: 10.22331/q-2023-09-11-1107. URL: https://doi.org/10.22331/q-2023-09-11-1107.

[11] Gregory D. Kahanamoku-Meyer et al. "Classically verifiable quantum advantage from a computational Bell test". In: *Nature Physics* 18.8 (Aug. 2022), pp. 918–924. ISSN: 1745-2481. DOI: 10.1038/s41567-022-01643-7. URL: http://dx.doi.org/10.1038/s41567-022-01643-7.

[12] Minzhao Liu et al. "Certified randomness using a trapped-ion quantum processor". In: *Nature* 640.8058 (Mar. 2025), pp. 343–348. ISSN: 1476-4687. DOI: 10.1038/s41586-025-08737-1. URL: http://dx.doi.org/10.1038/s41586-025-08737-1.

[13] Dan Shepherd and Michael J. Bremner. "Temporally unstructured quantum computation". In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 465.2105 (Feb. 2009), pp. 1413–1439. ISSN: 1471-2946. DOI: 10.1098/rspa.2008.0443. URL: http://dx.doi.org/10.1098/rspa.2008.0443.