Quantum Schur Sampling Circuits can be Strongly Simulated

Vojtěch Havlíček
Quantum Group, Department of Computer Science, University of Oxford,
Wolfson Building, Parks Road, Oxford OX1 3QD, UK

Sergii Strelchuk
Department of Applied Mathematics and Theoretical Physics,
University of Cambridge, Wilberforce Road, Cambridge, CB2 3HU, UK

Permutational Quantum Computing (PQC) ([Quantum Info. Comput., 10, 470–497, (2010)]) is a natural quantum computational model conjectured to capture non-classical aspects of quantum computation. An argument backing this conjecture was the observation that there was no efficient classical algorithm for estimation of matrix elements of the \( S_n \) irreducible representation matrices in the Young’s orthogonal form, which correspond to transition amplitudes of a broad class of PQC circuits. This problem can be solved with a PQC machine in polynomial time, but no efficient classical algorithm for the problem was previously known. Here we give a classical algorithm that efficiently approximates the transition amplitudes up to polynomial additive precision and hence solves this problem. We further extend our discussion to show that transition amplitudes of a broader class of quantum circuits – the Quantum Schur Sampling circuits – can be also efficiently estimated classically.

The effort for building quantum computers is driven by a belief that they can perform computational tasks beyond capabilities of classical computers. Despite a plethora of evidence for this conjecture, it remains unclear what kinds of quantum algorithms would lead to such computational advantage. With the development of nascent proof-of-principle quantum devices, there is an increased demand for simple and useful quantum protocols which could clearly demonstrate their supra-classical capabilities. This motivated various attempts to delineate computational power of the near-term devices in the regime where quantum error correction is not readily available [1–3]. One such direction is the study of restricted quantum computation that aims to identify sources of quantum advantage within computational models that are likely to be less powerful than universal quantum computation.

We study Permutational Quantum Computing (PQC) introduced by Marzuoli and Rasetti and further developed by Jordan [4, 5] – a natural quantum computational model that is conjectured to capture non-classical aspects of quantum computation. The backing for this conjecture came from several computational problems which were solvable on a PQC machine in polynomial time but the only known classical algorithms required exponential runtime. We provide an efficient classical probabilistic algorithm for polynomially small approximation of transition amplitudes of a class of PQC computations that allows us to resolve one of the problems used to back up the conjecture of the model supra-classicality. While this does not fully resolve the computational power of the model, it should be seen as a step towards isolating features of the model that could be responsible for its quantum computational advantage. Furthermore, we extend this result to a larger class of important quantum circuits – the so-called Quantum Schur Sampling circuits – which get their name from bearing structural similarities to circuits that perform Quantum Fourier Sampling [6].

Following the applications of PQC outlined by Jordan [5], the proposed algorithm can be used not only to ascertain the power of the computational model but also to potentially address problems of practical relevance: it can be used to estimate non-trivial elements of irreducible representation matrices of the symmetric group on \( n \) elements in the Young-Yamanouchi basis [7], which could find applications in quantum chemistry and various quantum information protocols.

PQC took inspiration in Topological Quantum Computing; a proposal for quantum computation by anyonic braiding and measurement in which results of the computation are determined solely by the topology of anyon trajectories. This approach relies on the existence of non-abelian anyons which have not been experimentally confirmed and its viability thus remains unknown. Permutational Quantum Computing can be then seen as an attempt to mimic some of this structure in a qubit-based model, in which angular momentum (spin) eigenstates roughly play the role of anyons and the computation proceeds by qubit interchange and spin measurements. The analogy between the two models is however very loose, as PQC completely disregards topology by forgetting about anyonic trajectories and only accounts for particle permutations; hence its name.

PQC takes input in a basis defined by a set of spin measurements on qubit subsets. This basis, similarly to Fourier basis, is defined by outcomes of nonlocal measurements and is therefore not a tensor product basis. Nevertheless, there exists an efficient quantum algorithm to prepare the basis states. It is also possible that a
suitable sequence of spin measurements could be implemented directly experimentally [5].

We now describe the basis. Consider \( n \) two-level quantum systems or qubits. With a convention that \( \hbar = 1 \), a spin of \( k \)-th qubit is formally defined by a triple:

\[
\vec{s}_k = \frac{1}{2}(X_k, Y_k, Z_k),
\]

where \( X_k \) denotes a Pauli \( X \) operator applied to \( k \)-th qubit. The total spin of a subset of qubits \( A \) is then given by:

\[
S_A^2 = \left( \sum_{k \in A} \vec{s}_k \right)^2.
\]

Let \( Z_A := \frac{1}{2} \sum_{k \in A} Z_k \) denote the \( z \)-spin operator on a qubit subset \( A \) and let \( Z \) be such operator applied to all of the \( n \) qubits. The operators \( Z_A \) and \( S_A^2 \) commute and share eigenspace labeled by quantum numbers \( j_A \) and \( m_A \).

Here \( j_A \) is the total spin of qubits in \( A \) and \( m_A \) is the multiplicity label taking values in integer steps between \( -j_A \) and \( j_A \). The \( j \)-quantum numbers are either integer or half-integer and combine according to the angular momentum addition rules [8]:

\[
|j_A - j_B| \leq j_{A \cup B} \leq j_A + j_B, \quad j_{A \cup B} = j_A + j_B \in \mathbb{Z}.
\]

The operators \( S_A^2 \) and \( S_B^2 \) can be shown to commute if and only if \( A \) and \( B \) are disjoint or one is subset of the other. An orthonormal basis can be then specified by listing outcomes of a complete set of commuting spin measurements on qubit subsets. Specifically, a basis can be defined by coupling one qubit at a time - that is by joint eigenstates of operators \( S_{[01]}^2, S_{[012]}^2, S_{[0123]}^2, \ldots S^n \) along with the \( Z \) operator and will be referred to as \textit{sequentially coupled basis}. An example of a sequentially coupled basis state on three qubits would be:

\[
|J = \frac{1}{2}, M = \frac{1}{2}, j_{01} = 1 \rangle_{\text{sch}} = \sqrt{\frac{7}{3}} |010 \rangle - \frac{1}{\sqrt{6}} |100 \rangle.
\]

Since the basis change does not alter the Hilbert space dimension the basis states can be assigned a bitstring \( x \in \{0, 1\}^n \) and we will label the sequentially coupled basis state corresponding to a bitstring \( x \) by \( |x\rangle_{\text{sch}} \equiv |J, M, \ldots \rangle_{\text{sch}} \) where appropriate. Note that the \( j \) values label spaces arising in the decomposition of the \( n \)-qubit Hilbert space into a direct sum of SU(2) irreducible representations. Representation-theoretically, the sequentially coupled basis is the Young-Yamanouchi basis [5, 7].

The unitary transformation \( U_{\text{Sch}} \) between the computational and the sequentially coupled basis is called the Quantum Schur Transformation. Similarly to the Quantum Fourier Transform, it is a global transformation from the computational basis to the basis labeled by the spin eigenvalues [9]. Recall that states in two spin eigenspaces \( j_L \) and \( j_R \) couple to a spin eigenstate \(|j, m \rangle\) as:

\[
|j, m \rangle := \sum_{m_L, m_R} C_{j_L, m_L; j_R, m_R}^{j, m} |j_L, m_L \rangle \otimes |j_R, m_R \rangle,
\]

where \( C_{j_L, m_L; j_R, m_R}^{j, m} \) are the Clebsch-Gordan (CG) coefficients. Note also that one can identify the computational basis states with a tensor product of spin eigenstates by:

\[
\frac{1}{2} m = -\frac{1}{2} \equiv |0 \rangle, \quad \frac{1}{2} m = \frac{1}{2} \equiv |1 \rangle. \tag{1}
\]

Since the spin of a single qubit is \( j = \frac{1}{2} \), we will drop the \( j \) label for qubits where appropriate to reduce notational clutter. The Schur transform on \( n \) qubits \(|m_0 \rangle \ldots |m_{n-1} \rangle\) in the sequentially coupled basis starts by coupling \(|m_0 \rangle \) with \(|m_1 \rangle \) into \(|j_{01}, m_{01} \rangle \), and proceeds by coupling \(|m_2 \rangle \) with \(|j_{01}, m_{01} \rangle \) into \(|j_{012}, j_{01}, m_{012} \rangle \). This state still carries the label \( j_{01} \) since the total angular momentum operators \( S_{[012]} \) and \( S_{[01]} \) commute. In contrast, the label \( m_{01} \) is discarded, because \( Z_{[01]} \) no longer commutes with \( S_{[012]} \). This process is continued until the whole register is coupled into a sequentially coupled basis state \(|J, M, \ldots, j_{012} \rangle_{\text{sch}} \). The transformation can be efficiently implemented using the algorithms of Bacon, Harrow and Chuang and Kirby and Strauch (see [9–12] for details).

We now define the computational model. Given a permutation \( \pi \) on \( n \) qubit labels and a classical input \( x \in \{0, 1\}^n \):

1. Prepare a sequentially coupled basis state \( U_{\text{Sch}} |x\rangle = |x\rangle_{\text{sch}} \equiv |j_{01}, \ldots, J, M \rangle_{\text{sch}} \) by applying the Quantum Schur Transformation \( U_{\text{sch}} \) to a computational basis state \(|x\rangle\).

2. Execute the ‘program’ \( \pi \), which permutes the qubits (or equivalently, their labels). That is:

\[
|j_{01}, \ldots, J, M \rangle_{\text{sch}} \mapsto |j_{\pi(0)\pi(1)}, \ldots, J, M \rangle_{\text{sch}}.
\]

This transformation can be also regarded as a sequence of computational basis \textsc{swap} gates that has
depth $O(n^2)$ by bubblesort algorithm [5]. The operation acts as a unitary $U_*$ on the input state.

3. Apply the inverse Quantum Schur Transform $U_{\text{Sch}}^\dagger$ to get a superposition over output states $|y\rangle$.

4. Measure the $y \equiv J, M, j' \ldots$ labels.

After executing steps 1-4, the algorithm samples from a probability distribution $p_x(y|x)$ over the output $y$.

We now show that the transition amplitudes of Permutational Quantum Computing circuits that use the sequentially coupled basis can be efficiently approximated classically. It is known that the PQC transition amplitudes in this regime correspond to the matrix elements of the irrep. matrices of the symmetric group in Young’s orthogonal form, so resolving this problem provides a classical algorithm for task that has been conjectured hard in [5] (See [5] for additional details of the correspondence). Our proof strategy follows work of van den Nest by showing that the output state is computationally tractable and invoking a theorem about classical simulation of overlaps of such states [13].

**Definition 1** (Computational Tractability [13]). An $n$-qubit state $|\psi\rangle$ is computationally tractable if it is possible to classically efficiently sample from $p = \{|\psi\rangle^2 : y \in \{0,1\}^n\}$ and the overlaps $\langle y | \psi \rangle$ can be computed classically up to $m$ significant bits in time $\text{poly}(n, m)$ for any computational basis state $|y\rangle$.

We first prove that the state $\langle y | J, M, \ldots \rangle_{\text{Sch}}$ can be efficiently computed classically to the required precision. To simplify the presentation, we work with either three or four qubits – the method can be straightforwardly generalized to an arbitrary number of qubits. By angular momentum conservation, the CG coefficients satisfy:

$$C_{j_1, m_1; j_2, m_2}^{J, M} = (\frac{-1}{2})^{M-J} \sqrt{2J + 1} \begin{pmatrix} j_1 & j_2 & J \\ m_1 & m_2 & M \end{pmatrix},$$

where the six-symbol array denotes the Wigner 3j symbol. Since the total angular momenta ($j$-numbers) are all upper-bounded by $\frac{n}{2}$ by the angular momentum addition rules, the Wigner 3j symbols can be computed in polynomial time to exponential precision using the Racah formula [14]. This allows for efficient evaluation of the Clebsch-Gordan coefficients up to the $m$ significant bits in $\text{poly}(n, m)$ time (see the Supplemental Material for additional details). Thus, a product of $(n - 1)$ CG coefficients can be evaluated efficiently to the necessary precision.

We now prove the second computational tractability condition, that is that the probability distributions arising from computational basis measurements on the sequentially coupled states can be sampled classically in polynomial time. We first show that it is possible to efficiently compute the ‘telescoping’ marginals of the output probability distribution $p(y) := |\langle y | J, M, \ldots \rangle_{\text{Sch}}|^2$, which are defined by:

$$p(y_k \ldots y_{n-1}) = \sum_{y_0 \ldots y_{k-1}} p(y_0 \ldots y_{n-1}).$$

This can be achieved by successive application of:

$$p(y_k \ldots y_{n-1}) = \sum_{y_0 \ldots y_{k-1}} p(y_0 \ldots y_{n-1})$$

$$= \sum_{m_0 \ldots m_{k-1}} C_{j_1, m_1; m_2}^{J, M} \left| C_{j_0, m_0}^{J, M} \right|^2 \sum_{y_0 \ldots y_{k-1}} p(y_0 \ldots y_{n-2}),$$

where we again used the correspondence of Eq. 1 and the angular momentum conservation of Eq. 2 was used to proceed from the second to the third line - recall that $M$ is fixed as the basis states are eigenstates of $Z$ with this eigenvalue and that $m_{n-1}$ and hence $M - m_{n-1}$ is also fixed, since $m_{n-1}$ is known. In contrast, the $M_k$ were all $a \text{ priori}$ summed over. The substitution hence allows to pull a CG factor in front of the above summation. Repeating this $(n - k)$ times and noting that by normalization:

$$\sum_{y_0 \ldots y_{k-1}} p(y_0 \ldots y_{n-1}) = 1,$$

gives the desired marginal as a factor of at most $(n - 1)$ CG coefficients. Hence the telescoping marginals are also polytime classically computable to the desired precision.
The output distribution on n qubits can be then efficiently sampled from by using the chain rule for probability distributions:

\[ p(y_0 \ldots y_{n-1}) = p(y_0|y_1 \ldots y_{n-1}) \ldots p(y_{n-2}|y_{n-1})p(y_{n-1}), \]

which implies that the output probability distribution can be computed by successively drawing individual bits. First, sample \( y_{n-1} \), which can be done by flipping a coin with bias given by the Chernoff-Hoeffding bound on \( p(y_{n-1}) \). Fix \( y_{n-1} \) to the result, and generate a sample \( y_{n-2} \) by flipping a coin with bias \( p(y_{n-2}|y_{n-1}) \). Proceed the same way up to \( y_0 \). The resulting bitstring \( y = y_0y_1 \ldots y_{n-1} \) of the results is then drawn from \( p(y) = p(y_0y_1 \ldots y_{n-1}) \) by the chain rule, as desired. One can hence sample the output in polynomial time classically.

We can now prove that an output probability distribution of the Permutational Quantum Computing in the bases considered here can be approximated in polynomial time classically. Consider a Permutational Quantum Computing circuit of the form:

\[
p_\pi(y|x) = |\langle y_0 \ldots y_n | U_{\pi}^T U_{\pi} | J, M, j_{012}, j_{01r} \rangle_{Sch} |^2
\]

where \( U_{\pi} \) implements the permutation \( \pi \) on qubit indices. From the above, \( |\phi \rangle \) and \( |\psi \rangle \) are computationally tractable (this follows directly from Definition 1 by substituting \( |y\rangle \) with \( \pi |y\rangle \)) and yield probability distributions:

\[ p := |\langle y|\phi \rangle |^2, \quad p_\pi := |\langle y|\psi \rangle |^2. \]

As in Theorem 3 of [13], consider an indicator function:

\[ \Upsilon(x) = \begin{cases} 1 & \text{if } p(y) > p_\pi(y), \\ 0 & \text{if } p(y) \leq p_\pi(y), \end{cases} \]

and define:

\[ F(x) := \frac{\langle \phi|y \rangle \langle y|\psi \rangle}{p(y)} \Upsilon(y), \quad G(y) := \frac{\langle \phi|y \rangle \langle y|\psi \rangle}{p_\pi(y)}(1 - \Upsilon(y)). \]

Both of these functions can be evaluated efficiently to the required precision, since \( |\psi \rangle \) and \( |\phi \rangle \) are both computationally tractable. Then:

\[ \langle \phi|\psi \rangle = \sum_y p(y)F(y) + \sum_y p_\pi(y)G(y). \tag{3} \]

The overlaps \( \langle \phi|\psi \rangle \) are all real-valued as they are products of the CG coefficients, which are real. It follows there is a polynomially-precise additive approximation algorithm to the state overlaps by the Chernoff-Hoeffding bound:

**Definition 2** (Chernoff-Hoeffding bound [15]). Let \( K_0, K_1, \ldots, K_T \) be i.i.d. samples from a real random variable \( K \), such that \( |K_i| \leq 1 \). Let \( \langle K \rangle \) be the expectation value of \( K \). Then for \( T = \frac{2}{\delta^2} \log \left( \frac{1}{\epsilon^2} \right) \):

\[ Pr \left( \left| \sum_{k=0}^{T} K_k - \langle K \rangle \right| \leq \epsilon \right) \geq 1 - \delta. \]

This implies that \( O \left( \frac{1}{\delta^2} \log \left( \frac{1}{\epsilon^2} \right) \right) \) samples suffice to give a polynomially precise approximation to both expectation values \( \langle F \rangle := \sum_y p(y)F(y) \) and \( \langle G \rangle := \sum_y p_\pi(y)G(y) \) with the accuracy \( \epsilon \) and failure rate \( \delta \). Setting \( \epsilon = \frac{1}{\|p(y)\|} \) hence gives an efficient simulation algorithm with exponentially small failure rate \( \delta \).

The analysis can be further extended to the case when the quantum circuit between the two Schur transformations is a composition of the permutation \( \pi \) along with \( Z \)-diagonal circuit \( \Lambda \) with classically efficiently computable elements. The output probabilities in the computational basis then remain unchanged as the elements of \( \Lambda \) are pure phases:

\[
p(y) := |\langle y|\Lambda |J, M, \ldots \rangle_{Sch}|^2
\]

This implies that the distributions from which \( F, G \) are drawn in Eq. 3 remain unchanged - although \( F, G \) become complex-valued. The only difference this makes for the presented algorithm is that in definition 2, both real and complex part of the amplitude have to be estimated at the same time.

Such quantum circuits are structurally similar to Quantum Fourier Sampling circuits - except the Fourier transform is now replaced by the Schur transform.

This observation hints at the importance of considering the presented classical algorithm when devising new quantum algorithms or arguments for quantum computational advantage.

In this letter, we studied the computational power of the Permutational Quantum Computing and found an efficient classical simulation for a wide class of problems which are efficiently solvable within this model. This allowed us to find a solution to a problem of approximating matrix elements of the \( S_n \) irreps in the Young’s orthogonal form, for which there was previously no classical efficient solution [5].

We now discuss some open questions. While we presented method for computing transition amplitudes for a broad class of PQC circuits, finding a method for sampling their output distribution remains open.

As we studied computational power of PQC in a regime where the input/output states are encoded in the sequentially-coupled basis, it also remains open if our construction generalizes to arbitrary bases defined by
complete commuting sets of spin subset measurements as the telescopic marginal method does not generalize simply to this case.

It is also an open question to fully characterize admissible quantum gates that extend the notion of the ‘program’ $\pi$ in step 2 beyond permutations or $Z$-diagonal gates, such that the classical tractability of the resulting quantum state remains preserved. It would be additionally interesting to further explore connections to the proposal of a computational model based on permuting distinguishable quantum particles in superposition by Aaronson et al. [16], as PQC is one of the limiting cases of this model. Lastly, we note that most of the ideas behind the PQC model originated from the spin networks [4, 17]. It would be interesting to study further interplay of our result with this framework.

We acknowledge discussions with S.P. Jordan, S. Mehraban and G. de Felice. We are grateful to an anonymous referees for pointing out technical slip-ups in the previous version of the paper. S. Strelchuk is supported by a Leverhulme Trust Early Career Fellowship. V. Havlicek is supported by Clarendon and Keble de Breyne scholarships.

* vojtech.havlicek@keble.ox.ac.uk

[1] M. J. Bremner, R. Jozsa, and D. J. Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society of London Series A*, 467:459–472, February 2011.

[2] M. J. Bremner, A. Montanaro, and D. J. Shepherd. Average-Case Complexity Versus Approximate Simulation of Commuting Quantum Computations. *Physical Review Letters*, 117(8):080501, August 2016.

[3] A. W. Harrow and A. Montanaro. Quantum computational supremacy. *Nature*, 549:203 EP –, 09 2017.

[4] A. Marzuoli and M. Rasetti. Computing spin networks. *Annals of Physics*, 318:345–407, August 2005.

[5] S. P. Jordan. Permutational quantum computing. *Quantum Info. Comput.*, 10(5):470–497, May 2010.

[6] B. Fefferman and C. Umans. The Power of Quantum Fourier Sampling. *ArXiv e-prints*, July 2015.

[7] R. Pauncz. *Alternant Molecular Orbital Method*. Studies in physics and chemistry, no. 4. Saunders, 1967.

[8] P. Weit. *Quantum Theory, Groups and Representations: An Introduction (Final draft version)*. 2017.

[9] D. Bacon, I. L. Chuang, and A. W. Harrow. Efficient Quantum Circuits for Schur and Clebsch-Gordan Transforms. *Phys. Rev. Lett.*, 97:170502, Oct 2006.

[10] A. M. Childs, A. W. Harrow, and P. Wocjan. Weak Fourier-Schur Sampling, the Hidden Subgroup Problem, and the Quantum Collision Problem. In Wolfgang Thomas and Pascal Weil, editors, *STACS 2007*, pages 598–609, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[11] D. Bacon, I. L. Chuang, and A. W. Harrow. The Quantum Schur and Clebsch-Gordan Transforms: I. Efficient Qudit Circuits. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA ’07, pages 1235–1244, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[12] W. M. Kirby and F. W. Strauch. A Practical Quantum Algorithm for the Schur Transform. *ArXiv e-prints*, September 2017.

[13] M. Van Den Nest. Simulating quantum computers with probabilistic methods. *Quantum Info. Comput.*, 11(9-10):784–812, September 2011.

[14] Giulio Racah. Theory of complex spectra. ii. *Physical Review*, 62(9-10):438, 1942.

[15] M. Schwarz and M. Van den Nest. Simulating Quantum Circuits with Sparse Output Distributions. *ArXiv e-prints*, October 2013.

[16] S. Aaronson, A. Bouland, G. Kuperberg, and S. Mehraban. The Computational Complexity of Ball Permutations. *ArXiv e-prints*, October 2016.

[17] R. Penrose. Angular momentum: an approach to combinatorial space-time. In T. Bastin, editor, *Quantum Theory and Beyond*. Cambridge University Press, Cambridge, 1971.

[18] E. W. Weisstein. *Wigner 3j-Symbol*. MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/Wigner3j-Symbol.html.
Evaluation of the CG coefficients. Here we describe how to efficiently evaluate the CG coefficients classically to the exponential precision. The CG coefficients are related to the Wigner $3j$ symbol as follows:

$$C^J,M_{j_1,m_1;j_2,m_2} = (-1)^{M+j_1-j_2} \sqrt{2J+1} \begin{pmatrix} j_1 & j_2 & J \\ m_1 & m_2 & M \end{pmatrix}.$$

(4)

The Wigner $3j$ symbol can be computed using the Racah formula [14]:

$$\begin{pmatrix} a & b & c \\ d & e & f \end{pmatrix} = \sum_t (-1)^{t+a-b-f} \frac{\sqrt{\Delta(abc)(a+d)!(b+e)!(c+f)!}(a-d)!(b-e)!(c-f)!}{t!(c-b+t+d)!(c-a+t-e)!(a+b-c-t)!(a-d-t)!(b-t-e)!},$$

(5)

where $\Delta(abc)$ is the triangle coefficient given by:

$$\Delta(abc) = \frac{(a+b+c)!(a-b-c)!(-a+b+c)!}{(a+b+c+1)!},$$

and the summation $\sum_t$ runs over all $t$ for which the factorials in the expression are well-defined. It is known that there are $\nu+1$ such terms, where [18]:

$$\nu = \min \{ a \pm d, b \pm e, c \pm f, a+b-c, b+c-a, c+a-b \} .$$

In the context considered here $a, b \ldots f$ are total angular momenta of $n$ qubits - so it always holds that $a, b \ldots f \leq \frac{n}{2}$. Hence, the summation in Eq. 5 runs over $\text{poly}(n)$ terms. Each of these terms only requires evaluation of factorials, division and square root - each of these operations can be performed efficiently up to $m$ bits in $\text{poly}(m)$ time. Evaluation of the Clebsch-Gordan coefficient up to $m$ significant bits will then take $\text{poly}(n, m)$ time, as required by the simulation algorithm.