

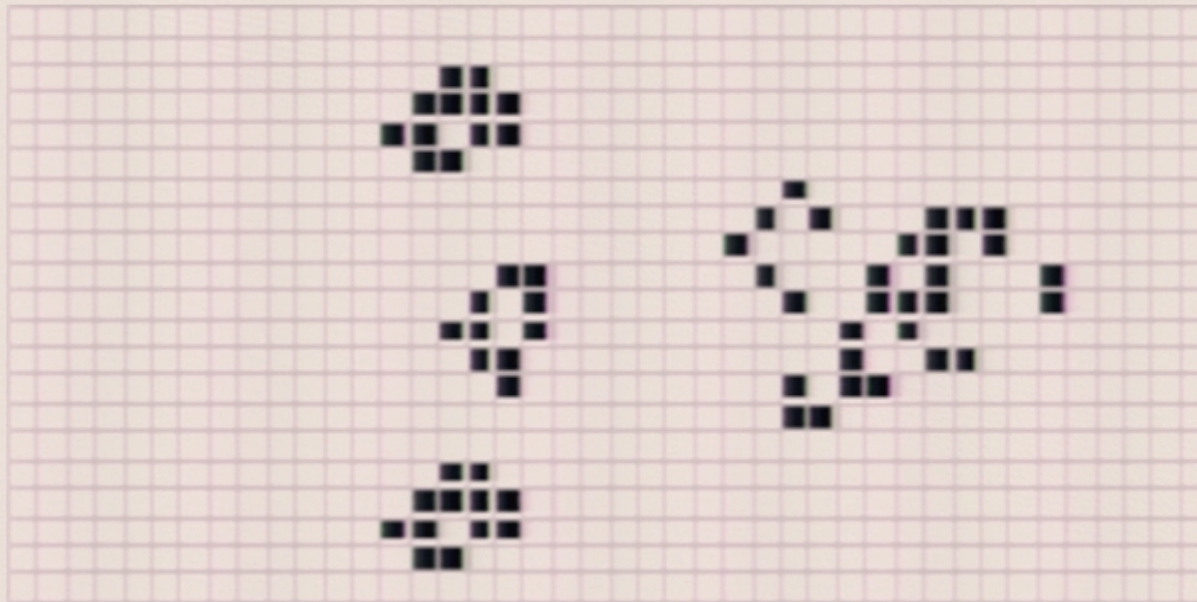
Title: Are Quantum States Exponentially Long Vectors?

Date: Jul 20, 2005 04:15 PM

URL: <http://pirsa.org/05070108>

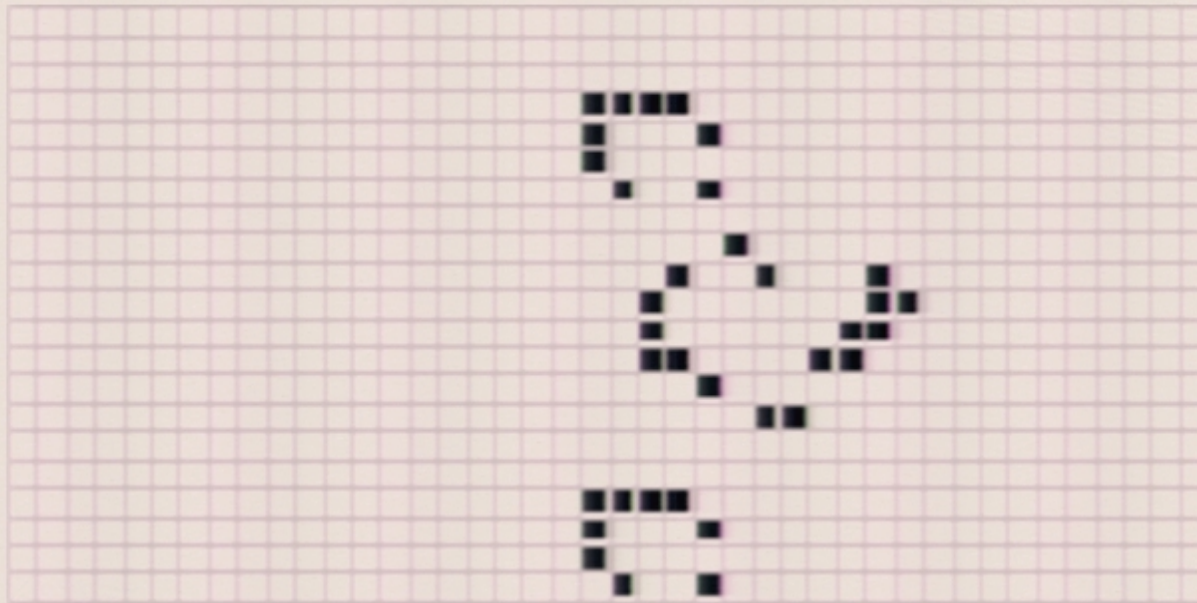
Abstract:

The Computer Science Picture of Reality



+ details

The Computer Science Picture of Reality



+ details

The Computer Science Picture of Reality



+ details

The Computer Science Picture of Reality



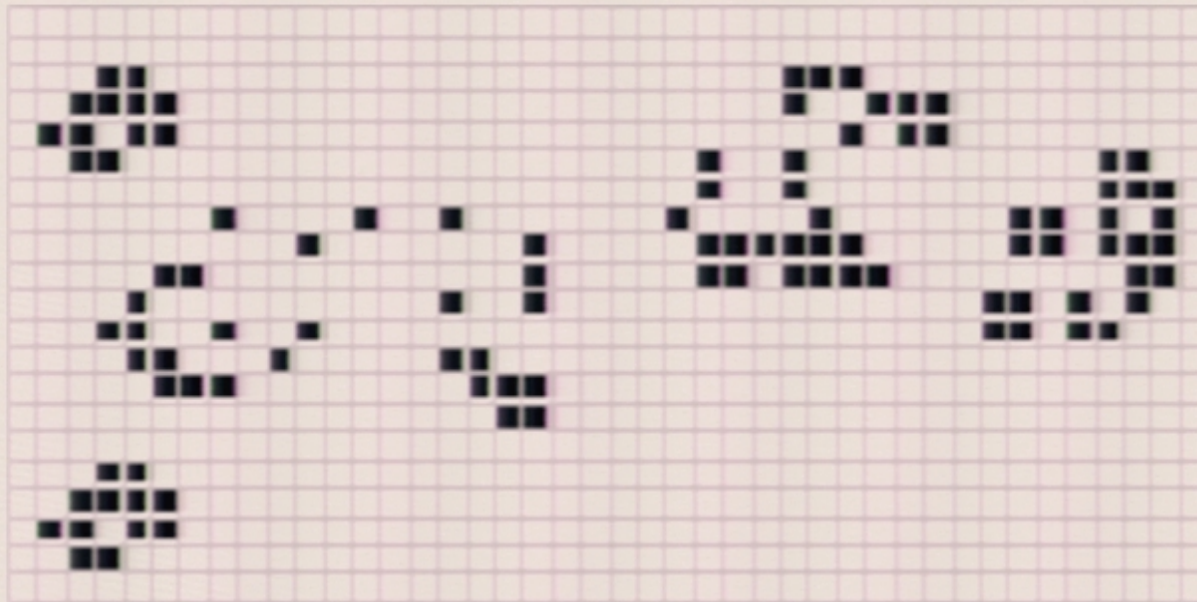
+ details

The Computer Science Picture of Reality



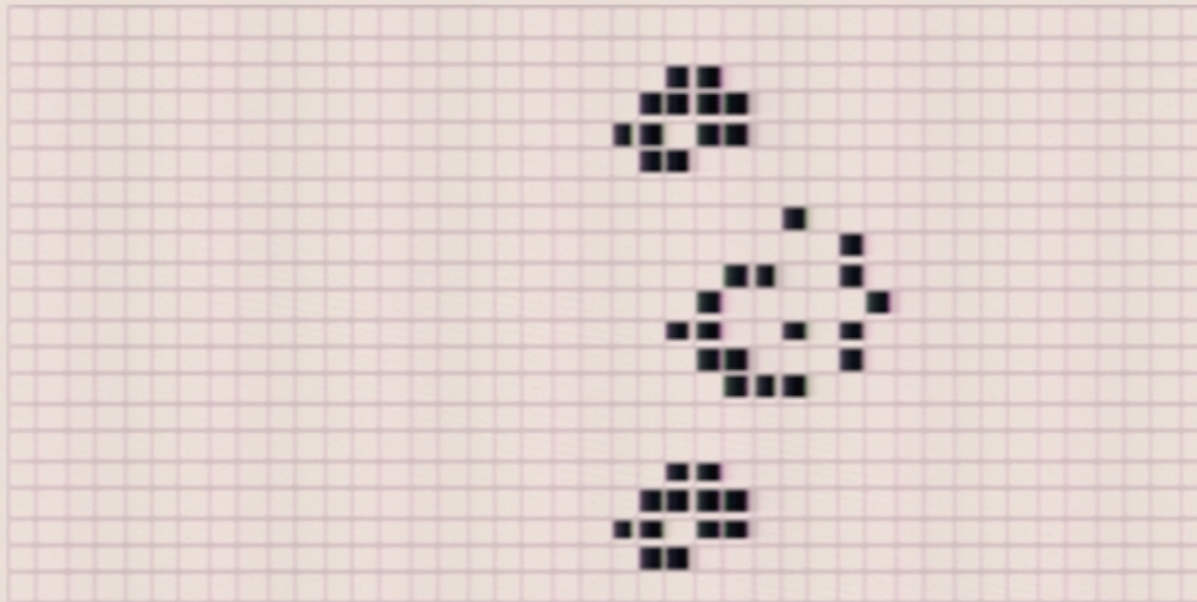
+ details

The Computer Science Picture of Reality



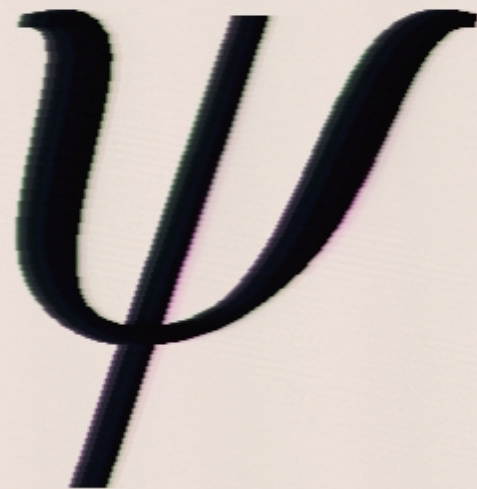
+ details

The Computer Science Picture of Reality



+ details

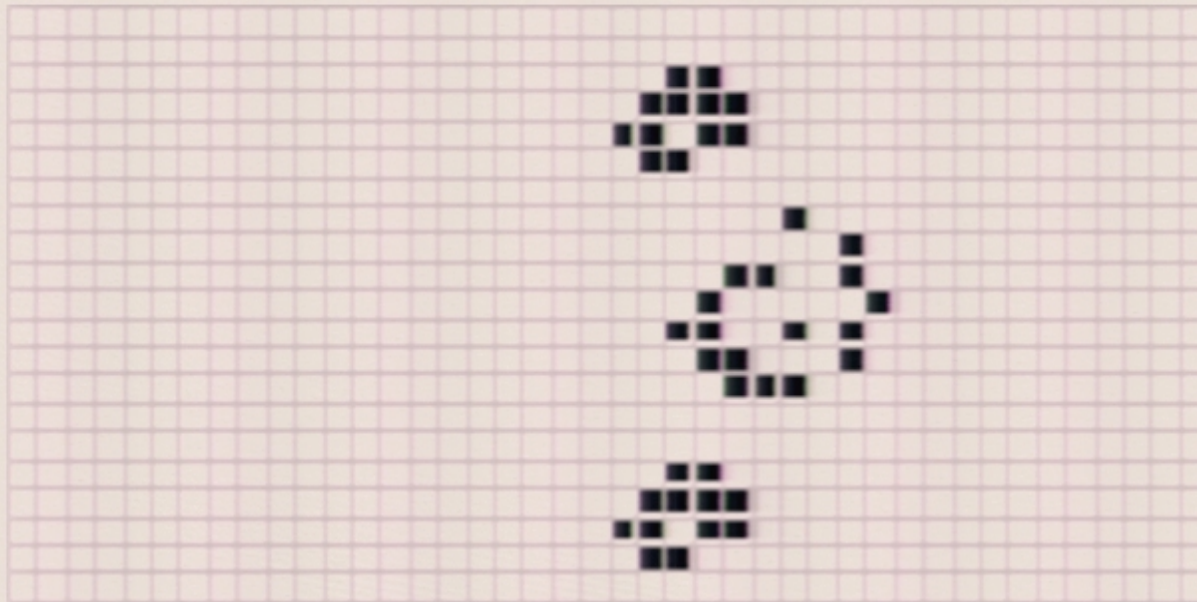
The Computer Science Picture of Reality

A large, bold, black Greek letter Psi (Ψ) symbol is centered on the slide. It is a stylized, calligraphic font with a thick stroke and a slight shadow effect.

Quantum computing challenges this picture

That's why everyone should care about it,
whether or not quantum factoring machines
are ever built

The Computer Science Picture of Reality



+ details

As far as I am concern[ed], the QC model consists of exponentially-long vectors (possible configurations) and some “uniform” (or “simple”) operations (computation steps) on such vectors ... The key point is that the associated complexity measure postulates that each such operation can be effected at unit cost (or unit time). My main concern is with this postulate. My own intuition is that the cost of such an operation or of maintaining such vectors should be linearly related to the amount of “non-degeneracy” of these vectors, where the “non-degeneracy” may vary from a constant to linear in the length of the vector (depending on the vector). Needless to say, I am not suggesting a concrete definition of “non-degeneracy,” I am merely conjecturing that such exists and that it capture[s] the inherent cost of the computation.

—Oded Goldreich



My Two-Pronged Response

My Two-Pronged Response

(1) It's not easy to explain current experiments
(let alone future ones!), if you don't think that quantum
states are exponentially long vectors

[A. 2004, "Multilinear Formulas and Skepticism of Quantum
Computing"]

My Two-Pronged Response

(1) It's not easy to explain current experiments
(let alone future ones!), if you don't think that quantum
states are exponentially long vectors

[A. 2004, "Multilinear Formulas and Skepticism of Quantum
Computing"]

(2) But it's not that bad

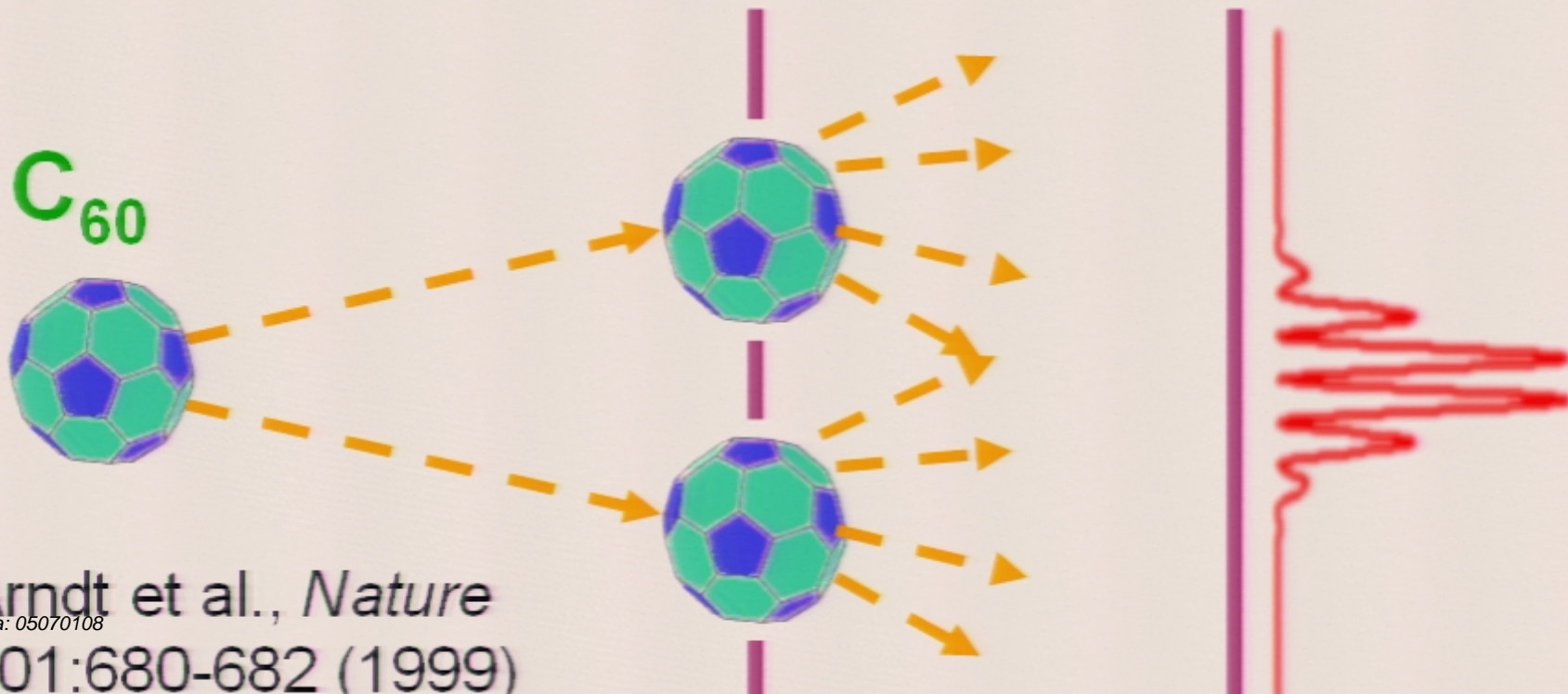
[A. 2004, "Limitations of Quantum Advice and One-Way
Communication"]

Prong (1)

Quantum states are exponentially long vectors

How Good Is The Evidence for QM?

- (1) **Interference:** Stability of e^- orbits, double-slit, etc.
- (2) **Entanglement:** Bell inequality, GHZ experiments
- (3) **Schrödinger cats:** C_{60} double-slit experiment, superconductivity, quantum Hall effect, etc.



Arndt et al., *Nature*
401:680-682 (1999)

Exactly what property separates
the **Sure States** we know we can
create, from the **Shor States** that
suffice for factoring?



DIVIDING LINE



Not precision in amplitudes:

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

Not precision in amplitudes:

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

Not entanglement across hundreds of particles:

$$\frac{|0\rangle^{\otimes n} + |1\rangle^{\otimes n}}{\sqrt{2}}$$

Not precision in amplitudes:

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

Not entanglement across hundreds of particles:

$$\frac{|0\rangle^{\otimes n} + |1\rangle^{\otimes n}}{\sqrt{2}}$$

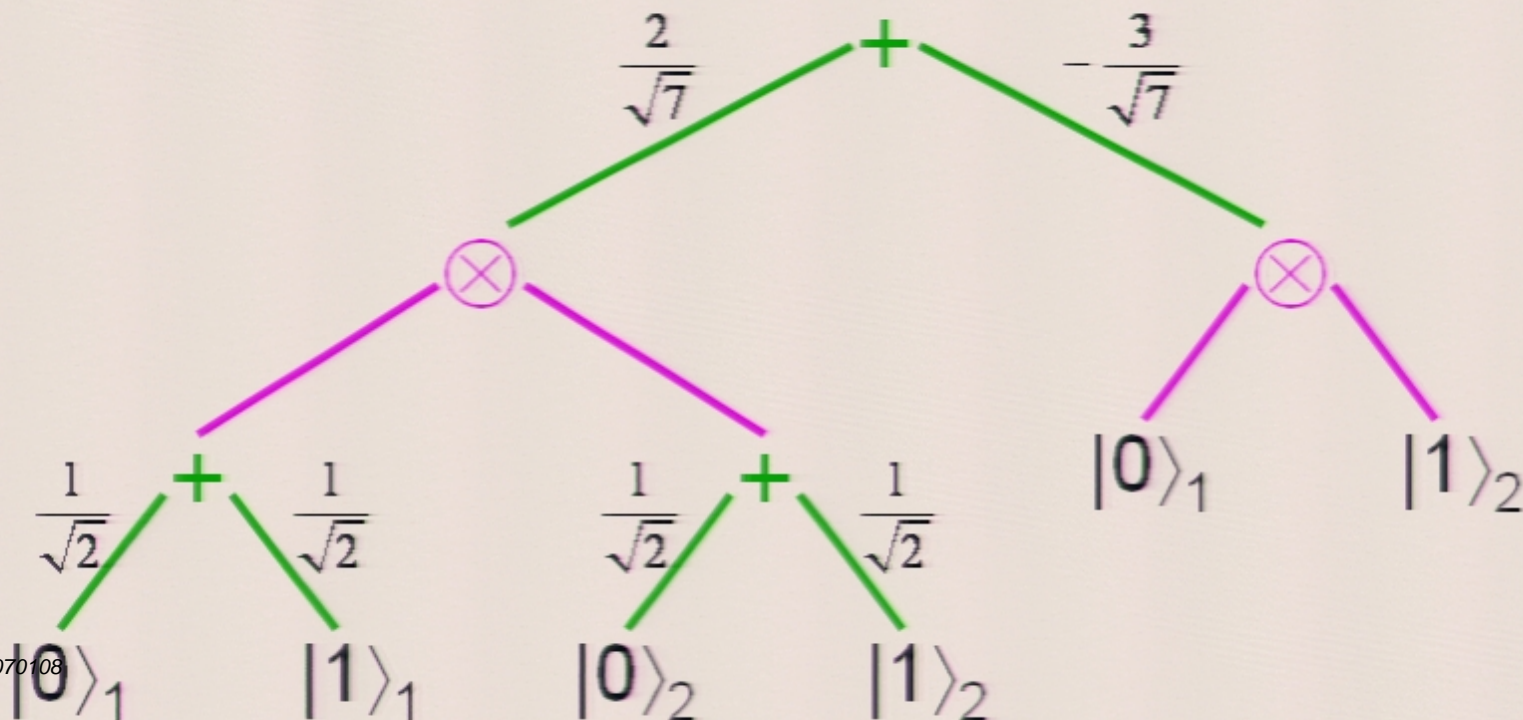
Not a combination of the two:

$$\frac{1}{\sqrt{2}} \left[\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right)^{\otimes n} + \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)^{\otimes n} \right]$$

Intuition: Once we accept $|\psi\rangle$ and $|\phi\rangle$ into our set of possible states, we're almost **forced** to accept $|\psi\rangle\otimes|\phi\rangle$ and $\alpha|\psi\rangle+\beta|\phi\rangle$ as well

Intuition: Once we accept $|\psi\rangle$ and $|\phi\rangle$ into our set of possible states, we're almost **forced** to accept $|\psi\rangle \otimes |\phi\rangle$ and $\alpha|\psi\rangle + \beta|\phi\rangle$ as well

But we might restrict ourselves to **tree states**: n-qubit states obtainable from $|0\rangle$ and $|1\rangle$ by a polynomial number of linear combinations and tensor products



Main Result

If $|C\rangle = \frac{1}{\sqrt{|C|}} \sum_{x \in C} |x\rangle$ is a uniform superposition

over the codewords of a binary linear code, then $|C\rangle$ requires tree size $n^{\Omega(\log n)}$ (even to approximate)—with high probability if the generator matrix is chosen uniformly at random

Prong (2)

It's not that bad

Prong (2)

It's not that bad

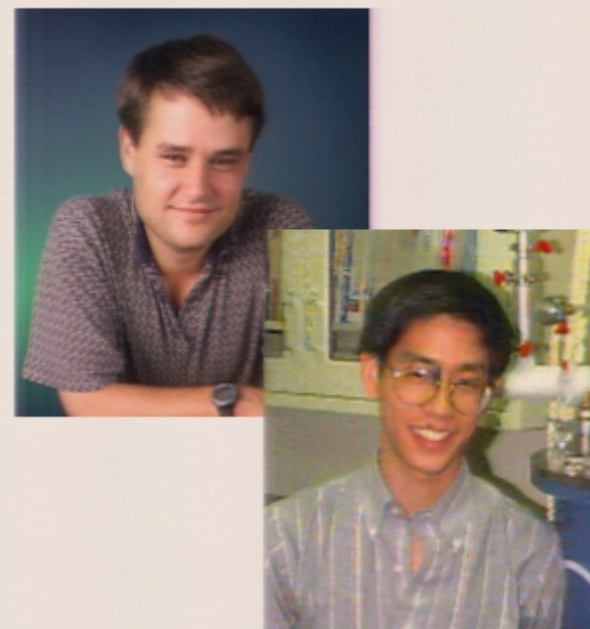
Quantum Advice

Nielsen & Chuang: “We know that many systems in Nature ‘prefer’ to sit in highly entangled states of many systems; might it be possible to exploit this preference to obtain extra computational power?”



Quantum Advice

Nielsen & Chuang: “We know that many systems in Nature ‘prefer’ to sit in highly entangled states of many systems; might it be possible to exploit this preference to obtain extra computational power?”



BQP/qpoly: Class of languages decidable by polynomial-size, bounded-error quantum circuits, given a polynomial-size **quantum advice state** $|\psi_n\rangle$ that depends only on the input length n

Challenge: Is quantum advice more powerful than classical advice?

I.e. does $BQP/qpoly$ strictly contain $BQP/poly$?

Challenge: Is quantum advice more powerful than classical advice?

I.e. does $BQP/qpoly$ strictly contain $BQP/poly$?

Notice that, given an **exponential** amount of classical advice, a computer could solve any problem whatsoever

So, could quantum advice be similar to that?
Could it let us solve, say, NP-complete problems in polynomial time?

PostBQP: Class of problems solvable in quantum polynomial time, if you can **postselect** measurement outcomes (i.e. measure, then kill yourself if you don't like the outcome)

A. 2004: PostBQP equals a classical complexity class called PP

PostBQP: Class of problems solvable in quantum polynomial time, if you can **postselect** measurement outcomes (i.e. measure, then kill yourself if you don't like the outcome)

A. 2004: PostBQP equals a classical complexity class called PP

Result: $BQP/qpoly \subseteq PostBQP/poly$

(Anything you can do with poly-size quantum advice, you can also do with poly-size *classical* advice, provided you're willing to use exponentially more computation time)

Intuition

Alice is the
“advisor”

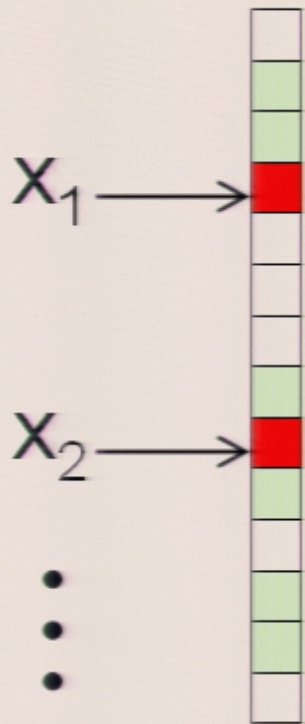


Bob is the
PostBQP
machine

Alice wanted to send Bob a Q-qubit quantum advice state $|\psi\rangle$

Instead she'll send Bob an $O(nQ)$ -qubit **classical** message, containing a “Darwinian training set” of inputs that he can use to reconstruct $|\psi\rangle$ on his own

Alice's Classical Message



Bob, if you use the maximally mixed state in place of my quantum message, then x_1 is the lexicographically first input for which you'll output the wrong answer with probability at least $1/3$.

But if you condition on succeeding on x_1 , then x_2 is the next input for which you'll output the wrong answer with probability at least $1/3$.

But if you condition on succeeding on x_1 and x_2 , then x_3 is the ...



Technicality: We assume Alice's quantum message was boosted, so that the error probability is negligible

Technicality: We assume Alice's quantum message was boosted, so that the error probability is negligible

Claim: Alice only needs to send $T=O(Q)$ inputs X_1, \dots, X_T

Proof Sketch: Bob succeeds on x_1, \dots, x_T simultaneously with probability at most $(2/3)^T$.

But we can decompose the Q -qubit maximally mixed state as $\frac{1}{2^Q} \sum_{i=1}^{2^Q} |\psi_i\rangle\langle\psi_i|$ where $|\psi_1\rangle = |\psi\rangle$ is

Alice's "true" quantum message. Therefore Bob succeeds with probability $\Omega(1/2^Q)$

Recent Improvement:

$BQP/qpoly \subseteq QMA/poly$

Here QMA (Quantum Merlin-Arthur) is the quantum generalization of NP: the class of problems for which a “yes” answer can be proven with a polynomial-size quantum witness

(Note that $QMA \subseteq PostBQP$)

Recent Improvement:

$$\text{BQP/qpoly} \subseteq \text{QMA/poly}$$

Here QMA (Quantum Merlin-Arthur) is the quantum generalization of NP: the class of problems for which a “yes” answer can be proven with a polynomial-size quantum witness

(Note that $\text{QMA} \subseteq \text{PostBQP}$)

Yet another improvement:

$\text{QCMA/qpoly} \subseteq \text{QMA/poly}$, where QCMA (Quantum Classical Merlin-Arthur) is the generalization of NP with a *quantum* verifier but a *classical* witness

[0-1-NP_C](#) - [1NAuxPDA^p](#) - [#AC⁰](#) - [#L](#) - [#L/poly](#) - [#GA](#) - [#P](#) - [#W\[t\]](#) - [EXP](#) - [L](#) - [L/poly](#) - [P](#) - [SAC⁰](#) - [SAC¹](#) - [A₀PP](#) - [AC](#) - [AC⁰](#) - [AC¹](#) - [AC⁰\[m\]](#) - [ACC⁰](#) - [AH](#) - [AL](#) - [AlgP/poly](#) - [AM](#) - [AM-EXP](#) - [AM intersect coAM](#) - [AM\[polylog\]](#) - [AmpMP](#) - [AmpP-BQP](#) - [AP](#) - [APP](#) - [APX](#) - [AUC-SPACE\(f\(n\)\)](#) - [AuxPDA](#) - [AVBPP](#) - [AvE](#) - [AvP](#) - [AW\[P\]](#) - [AWPP](#) - [AW\[SAT\]](#) - [AW\[*\]](#) - [AW\[t\]](#) - [AxP](#) - [AxPP](#) - [βP](#) - [BH](#) - [BPE](#) - [BPEE](#) - [BP_HSPACE\(f\(n\)\)](#) - [BPL](#) - [BP-NP](#) - [BPP](#) - [BPP^{cc}](#) - [BPP^{KT}](#) - [BPP//log](#) - [BPP-OBDD](#) - [BPP_{path}](#) - [BPQP](#) - [BPSPACE\(f\(n\)\)](#) - [BPTIME\(f\(n\)\)](#) - [BQNC](#) - [BQNP](#) - [BQP](#) - [BQP/log](#) - [BQP/poly](#) - [BQP/qlog](#) - [BQP/qpoly](#) - [BQP-OBDD](#) - [BQPSPACE](#) - [BQP_{tt}/poly](#) - [BQTIME\(f\(n\)\)](#) - [k-BWBP](#) - [C₋AC⁰](#) - [C₋L](#) - [C₋P](#) - [CFL](#) - [CLOG](#) - [CH](#) - [Check](#) - [C_kP](#) - [CNP](#) - [coAM](#) - [coC₋P](#) - [cofrIP](#) - [Coh](#) - [coMA](#) - [coMod_kP](#) - [complP](#) - [compNP](#) - [coNE](#) - [coNEXP](#) - [coNL](#) - [coNP](#) - [coNP^{cc}](#) - [coNP/poly](#) - [coNQP](#) - [coRE](#) - [coRNC](#) - [coRP](#) - [coSL](#) - [coUCC](#) - [coUP](#) - [CP](#) - [CSIZE\(f\(n\)\)](#) - [CSL](#) - [CZK](#) - [D#P](#) - [DCFL](#) - [Δ₂P](#) - [δ-BPP](#) - [δ-RP](#) - [DET](#) - [DiffAC⁰](#) - [DisNP](#) - [DistNP](#) - [DP](#) - [DQP](#) - [DSPACE\(f\(n\)\)](#) - [DTIME\(f\(n\)\)](#) - [DTISP\(t\(n\),s\(n\)\)](#) - [Dyn-FO](#) - [Dyn-ThC⁰](#) - [E](#) - [EE](#) - [EEE](#) - [EESPACE](#) - [EEXP](#) - [EH](#) - [ELEMENTARY](#) - [EL_kP](#) - [EPTAS](#) - [k-EQBP](#) - [EQP](#) - [EQTIME\(f\(n\)\)](#) - [ESPACE](#) - [ExistsBPP](#) - [ExistsNISZK](#) - [EXP](#) - [EXP/poly](#) - [EXPSPACE](#) - [FBQP](#) - [Few](#) - [FewP](#) - [FH](#) - [FNL](#) - [FNL/poly](#) - [FNP](#) - [FO\(t\(n\)\)](#) - [FOLL](#) - [FP](#) - [FP^{NP\[log\]}](#) - [FPR](#) - [FPRAS](#) - [FPT](#) - [FPT_{nu}](#) - [FPT_{su}](#) - [FPTAS](#) - [FQMA](#) - [frIP](#) - [F-TAPE\(f\(n\)\)](#) - [F-TIME\(f\(n\)\)](#) - [GA](#) - [GAN-SPACE\(f\(n\)\)](#) - [GapAC⁰](#) - [GapL](#) - [GapP](#) - [GC\(s\(n\),C\)](#) - [GCSL](#) - [GI](#) - [GLO](#) - [GPCD\(r\(n\),q\(n\)\)](#) - [G\[t\]](#) - [HeurBPP](#) - [HeurBPTIME\(f\(n\)\)](#) - [H_kP](#) - [HP](#) - [HVSZK](#) - [IC\[log.poly\]](#) - [IP](#) - [IPP](#) - [IP\[polylog\]](#) - [L](#) - [LIN](#) - [L_kP](#) - [LOGCFL](#) - [LogFew](#) - [LogFewNL](#) - [LOGNP](#) - [LOGSNP](#) - [L/poly](#) - [LWPP](#) - [MA](#) - [MA'](#) - [MAC⁰](#) - [MA-E](#) - [MA-EXP](#) - [mAL](#) - [MaxNP](#) - [MaxPB](#) - [MaxSNP](#) - [MaxSNP⁰](#) - [mcoNL](#) - [MinPB](#) - [MIP](#) - [MIP*\[2.1\]](#) - [MIP_{EXP}](#) - [\(M_k\)P](#) - [mL](#) - [mNC¹](#) - [mNL](#) - [mNP](#) - [Mod_kL](#) - [Mod_kP](#) - [ModP](#) - [ModZ_kL](#) - [mP](#) - [MP](#) - [MPC](#) - [mP/poly](#) - [mTC⁰](#) - [NAuxPDA^p](#) - [NC](#) - [NC⁰](#) - [NC¹](#) - [NC²](#) - [NE](#) - [NE/poly](#) - [Nearly-P](#) - [NEE](#) - [NEEE](#) - [NEEXP](#) - [NEXP](#) - [NEXP/poly](#) - [NIQSZK](#) - [NISZK](#) - [NISZK_h](#) - [NL](#) - [NL/poly](#) - [NLIN](#) - [NLOG](#) - [NP](#) - [NPC](#) - [NP^{cc}](#) - [NP_C](#) - [NPI](#) - [NP intersect coNP](#) - [\(NP intersect coNP\)/poly](#) - [NP/log](#) - [NPMV](#) - [NPMV-sel](#) - [NPMV_t](#) - [NPMV_t-sel](#) - [NPO](#) - [NPOPB](#) - [NP/poly](#) - [\(NP,P-samplable\)](#) - [NP_R](#) - [NPSPACE](#) - [NPSV](#) - [NPSV-sel](#) - [NPSV_t](#) - [NPSV_t-sel](#) - [NQP](#) - [NSPACE\(f\(n\)\)](#) - [NT](#) - [NTIME\(f\(n\)\)](#) - [OCQ](#) - [OptP](#) - [P](#) - [P/log](#) - [P/poly](#) - [P^{#P}](#) - [P^{#P\[1\]}](#) - [PAC⁰](#) - [PBP](#) - [k-PBP](#) - [P_C](#) - [P^{cc}](#) - [PCD\(r\(n\),q\(n\)\)](#) - [P-close](#) - [PCP\(r\(n\),q\(n\)\)](#) - [PermUP](#) - [PEXP](#) - [PF](#) - [PFCHK\(t\(n\)\)](#) - [PH](#) - [PH^{cc}](#) - [Φ₂P](#) - [PhP](#) - [Π₂P](#) - [PINC](#) - [PIO](#) - [P^k](#) - [PKC](#) - [PL](#) - [PL₁](#) - [PL_{infinity}](#) - [PLF](#) - [PLL](#) - [PLS](#) - [p^{NP}](#) - [p^{IINP}](#) - [p^{NP\[k\]}](#) - [p^{NP\[log\]}](#) - [p^{NP\[log²\]}](#) - [P-OBDD](#) - [PODN](#) - [polyL](#) - [PostBQP](#) - [PP](#) - [PP/poly](#) - [PPA](#) - [PPAD](#) - [PPADS](#) - [PPP](#) - [P^{PP}](#) - [PPSPACE](#) - [PQUERY](#) - [PR](#) - [P_R](#) - [Pr_HSPACE\(f\(n\)\)](#) - [PromiseBPP](#) - [PromiseBQP](#) - [PromiseP](#) - [PromiseRP](#) - [PrSPACE\(f\(n\)\)](#) - [P-Sel](#) - [PSK](#) - [PSPACE](#) - [PT₁](#) - [PTAPE](#) - [PTAS](#) - [PTWK\(f\(n\),g\(n\)\)](#) - [PZK](#) - [QAC⁰](#) - [QAC⁰\[m\]](#) - [QACC⁰](#) - [QAC_f⁰](#) - [QAM](#) - [QCFL](#) - [QCMa](#) - [QH](#) - [QIP](#) - [QIP\[2\]](#) - [QMA](#) - [QMA+](#) - [QMA\(2\)](#) - [QMA_{log}](#) - [QMAM](#) - [QMIP](#) - [QMIP_{le}](#) - [QMIP_{ne}](#) - [QNC](#) - [QNC⁰](#) - [QNC_f⁰](#) - [QNC¹](#) - [QP](#) - [QPLIN](#) - [QPSPACE](#) - [QRG](#) - [QS₂P](#) - [QSZK](#) - [R](#) - [RE](#) - [REG](#) - [RevSPACE\(f\(n\)\)](#) - [RG](#) - [RG\(1\)](#) - [R_HL](#) - [R_HSPACE\(f\(n\)\)](#) - [RL](#) - [RNC](#) - [RP](#) - [RPP](#) - [RSPACE\(f\(n\)\)](#) - [S₂P](#) - [S₂-EXP·P^{NP}](#) - [SAC](#) - [SAC⁰](#) - [SAC¹](#) - [SAPTIME](#) - [SBP](#) - [SC](#) - [SE](#) - [SEH](#) - [SelfNP](#) - [SF_k](#) - [Σ₂P](#) - [SKC](#) - [SL](#) - [SLICEWISE PSPACE](#) - [SNP](#) - [SO-E](#) - [SP](#) - [SP](#) - [span-P](#) - [SPARSE](#) - [SP^{NP}](#) - [SQG](#) - [SUBEXP](#) - [symP](#) - [SZK](#) - [SZK_h](#) - [TALLY](#) - [TC⁰](#) - [TFNP](#) - [Θ₂P](#) - [TreeBQP](#) - [TREE-REGULAR](#) - [UP](#) - [UCC](#) - [UE](#) - [UL](#) - [JL/poly](#) - [UP](#) - [US](#) - [VNC_k](#) - [VNP_k](#) - [VP_k](#) - [VQP_k](#) - [W\[1\]](#) - [WAPP](#) - [W\[P\]](#) - [WPP](#) - [W\[SAT\]](#) - [W\[*\]](#) - [W\[t\]](#)

0-1-NP_C - 1NAuxPDA^P - #AC⁰ - #L - #L/poly - #GA - #P - #W[t] - EXP - L - L/poly - P - SAC⁰ - SAC¹ - A₀PP
 - AC - AC⁰ - AC¹ - AC⁰[m] - ACC⁰ - AH - AL - AlgP/poly - AM - AM-EXP - AM - polylog] - AmpMP
 AmpP-BQP - AP - APP - APX - AUC-SPACE(f(n)) - AuxPDA - AVBPP - AWPP - AW[SAT]
 AW[*] - AW[t] - AxP - AxPP - βP - BH - BPF - BPF^{EXP} - BPF^{SPACE}(f(n)) - BPP^{CC} - BPP^{KT}
 BPP//log - BPP-OBDD - BPP_{path} - BPP_{poly} - BQP - BQP/log
 BQP/poly - BQP/qlog - BQP/qpoly - BPP - C_{AC}⁰ - C_L
 C_P - CFL - CLOG - CH - Check - compIP - compNP
 coNE - coNEXP - coNL - CSIZE(f(n)) - CSL - CTIME
 DSPACE(f(n)) - DTIME - ELEMENTARY - EL_k - EPTA
 EXP/poly - EXPSPACE - FBQP - FPRAS - FPT - FPT_{poly}
 FPRAS - FPT - FPT_{poly} - GapAC⁰ - G₁ - G₂ - GC₁
 H_kP - HP - H_k - log.poly - LOGSNP - L/poly - MAC⁰
 mcoNL - MinPB - MIP - MIP^{EXP} - (MIP^{EXP})_k - ModP - ModZ_kL
 mP - MP - MPC - mP/poly - N₁AuxPDA^P - NC⁰ - NC¹ - NE - NE₁
 NEEXP - NEXP - NEXP/poly - NSZK - NI₁ - NSZK_h - NL - NLIN - NPC - NPC^{CC} - NP_C
 NPI - NP intersect coNP - NP intersect coNP_{poly} - NP/log - NPMV_t-sel - NPMV_t-sel - NPO
 NPOPB - NP/poly - (NP.P₁solvable) - NPSPACE - NPSV_t-sel - NPSV_t-sel - NQP
 NSPACE(f(n)) - NT - NTIME - OCQ - Op_{log} - P/poly - #P[1] - PAC - k-PBP - P_C - P^{CC}
 PCD(r(n),q(n)) - P-close - P₁(n) - Perm - P - PF - PFC - PH - PH₁ - P - PhP - Π₂P - PINC
 - PIO - P^K - PKC - PL - PL₁ - PL_{infinity} - PLF - PLL - PLS - p^{NP} - p^{INP} - p^{NP[log]} - P-OBDD - PODN
 polyL - PostBQP - PP - PP/poly - PPA - PPAD - PPADS - PPP - PSPACE - QUERY - PR - P_R
 Pr_HSPACE(f(n)) - PromiseBPP - PromiseBQP - PromiseP - PromiseR₁SPACE(f(n)) - Sel - PSK - PSPACE
 PT₁ - PTAPE - PTAS - PTWK(f(n),g(n)) - PZK - QAC⁰ - QAC⁰[m] - Q₁ - QCFL - QCMA - QH
 QIP - QIP[2] - QMA - QMA+ - QMA(2) - QMA_{log} - QMAM - QMIP - QMIP_{le} - QMIP_{ne} - QNC⁰ - QNC_f⁰ - QNC¹
 QP - QPLIN - QPSPACE - QRG - QS₂P - QSZK - R - RE - REG - RevSPACE(f(n)) - RG - RG(1) - R_HL
 R_HSPACE(f(n)) - RL - RNC - RP - RPP - RSPACE(f(n)) - S₂P - S₂-EXP·P^{NP} - SAC - SAC⁰ - SAC¹ - SAPTIME - SBP
 SC - SE - SEH - SelfNP - SF_k - Σ₂P - SKC - SL - SLICEWISE PSPACE - SNP - SO-E - SP - SP - span-P - SPARSE
 PIRSA: 05070108 - SQG - SUBEXP - symP - SZK - SZK_h - TALLY - TC⁰ - TFNP - Θ₂P - TreeBQP - TREE-REGULAR - AP
 - UCC - UE - UL - UL/poly - UP - US - VNC_k - VNP_k - VP_k - VQP_k - W[1] - WAPP - W[P] - WPP - W[SAT] - W[*] - W[t]



ALL
 = PostBQP/qpoly
 = QIP/qpoly

0-1-NP_C - 1NAuxPDA^P - #AC⁰ - #L - #L/poly - #GA - #P - #W[t] - EXP - L - L/poly - P - SAC⁰ - SAC¹ - A₀PP
 - AC - AC⁰ - AC¹ - AC⁰[m] - ACC⁰ - AH - AL - AlgP/poly - AM - AM-EXP - AM - polylog] - AmpMP
 AmpP-BQP - AP - APP - APX - AUC-SPACE(f(n)) - AuxPDA - AVBPP - AWPP - AW[SAT]
 AW[*] - AW[t] - AxP - AxPP - βP - BH - BPF - BPF^ε - BPP - BPP^{cc} - BPP^{KT}
 BPP//log - BPP-OBDD - BPP_{path} - BPP_{poly} - BQP - BQP/log
 BQP/poly - BQP/qlog - BQP/qpoly - BPP - C_AC⁰ - C_L
 C_P - CFL - CLOG - CH - Check - compIP - compNP
 coNE - coNEXP - coNL - CSIZE(f(n)) - CSL - CTIME
 DSPACE(f(n)) - DTIME - ELEMENTARY - EL_k - EPTA
 EXP/poly - EXPSPACE - FBQP - FPRAS - FPT - FPT_{poly} - GAN-SPACE(f(n)) - GAN-SPACE(f(n))
 GapAC⁰ - G₁ - G₂ - GC - H_kP - HP - H_k - H_k - log.poly - LOGSNP - L/poly - L_kWPP - MAC⁰
 mcoNL - MinPB - MIP - MIP^{EXP} - (MIP^{EXP})_{poly} - ModP - ModZ_kL
 mP - MP - MPC - mP/poly - mP/poly - mP/poly - NAuxPDA^P - NC⁰ - NC¹ - NE - NE₁ - NE₂ - NE₃ - NE₄ - NE₅ - NE₆ - NE₇ - NE₈ - NE₉ - NE₁₀ - NE₁₁ - NE₁₂ - NE₁₃ - NE₁₄ - NE₁₅ - NE₁₆ - NE₁₇ - NE₁₈ - NE₁₉ - NE₂₀ - NE₂₁ - NE₂₂ - NE₂₃ - NE₂₄ - NE₂₅ - NE₂₆ - NE₂₇ - NE₂₈ - NE₂₉ - NE₃₀ - NE₃₁ - NE₃₂ - NE₃₃ - NE₃₄ - NE₃₅ - NE₃₆ - NE₃₇ - NE₃₈ - NE₃₉ - NE₄₀ - NE₄₁ - NE₄₂ - NE₄₃ - NE₄₄ - NE₄₅ - NE₄₆ - NE₄₇ - NE₄₈ - NE₄₉ - NE₅₀ - NE₅₁ - NE₅₂ - NE₅₃ - NE₅₄ - NE₅₅ - NE₅₆ - NE₅₇ - NE₅₈ - NE₅₉ - NE₆₀ - NE₆₁ - NE₆₂ - NE₆₃ - NE₆₄ - NE₆₅ - NE₆₆ - NE₆₇ - NE₆₈ - NE₆₉ - NE₇₀ - NE₇₁ - NE₇₂ - NE₇₃ - NE₇₄ - NE₇₅ - NE₇₆ - NE₇₇ - NE₇₈ - NE₇₉ - NE₈₀ - NE₈₁ - NE₈₂ - NE₈₃ - NE₈₄ - NE₈₅ - NE₈₆ - NE₈₇ - NE₈₈ - NE₈₉ - NE₉₀ - NE₉₁ - NE₉₂ - NE₉₃ - NE₉₄ - NE₉₅ - NE₉₆ - NE₉₇ - NE₉₈ - NE₉₉ - NE₁₀₀ - NE₁₀₁ - NE₁₀₂ - NE₁₀₃ - NE₁₀₄ - NE₁₀₅ - NE₁₀₆ - NE₁₀₇ - NE₁₀₈ - NE₁₀₉ - NE₁₁₀ - NE₁₁₁ - NE₁₁₂ - NE₁₁₃ - NE₁₁₄ - NE₁₁₅ - NE₁₁₆ - NE₁₁₇ - NE₁₁₈ - NE₁₁₉ - NE₁₂₀ - NE₁₂₁ - NE₁₂₂ - NE₁₂₃ - NE₁₂₄ - NE₁₂₅ - NE₁₂₆ - NE₁₂₇ - NE₁₂₈ - NE₁₂₉ - NE₁₃₀ - NE₁₃₁ - NE₁₃₂ - NE₁₃₃ - NE₁₃₄ - NE₁₃₅ - NE₁₃₆ - NE₁₃₇ - NE₁₃₈ - NE₁₃₉ - NE₁₄₀ - NE₁₄₁ - NE₁₄₂ - NE₁₄₃ - NE₁₄₄ - NE₁₄₅ - NE₁₄₆ - NE₁₄₇ - NE₁₄₈ - NE₁₄₉ - NE₁₅₀ - NE₁₅₁ - NE₁₅₂ - NE₁₅₃ - NE₁₅₄ - NE₁₅₅ - NE₁₅₆ - NE₁₅₇ - NE₁₅₈ - NE₁₅₉ - NE₁₆₀ - NE₁₆₁ - NE₁₆₂ - NE₁₆₃ - NE₁₆₄ - NE₁₆₅ - NE₁₆₆ - NE₁₆₇ - NE₁₆₈ - NE₁₆₉ - NE₁₇₀ - NE₁₇₁ - NE₁₇₂ - NE₁₇₃ - NE₁₇₄ - NE₁₇₅ - NE₁₇₆ - NE₁₇₇ - NE₁₇₈ - NE₁₇₉ - NE₁₈₀ - NE₁₈₁ - NE₁₈₂ - NE₁₈₃ - NE₁₈₄ - NE₁₈₅ - NE₁₈₆ - NE₁₈₇ - NE₁₈₈ - NE₁₈₉ - NE₁₉₀ - NE₁₉₁ - NE₁₉₂ - NE₁₉₃ - NE₁₉₄ - NE₁₉₅ - NE₁₉₆ - NE₁₉₇ - NE₁₉₈ - NE₁₉₉ - NE₂₀₀ - NE₂₀₁ - NE₂₀₂ - NE₂₀₃ - NE₂₀₄ - NE₂₀₅ - NE₂₀₆ - NE₂₀₇ - NE₂₀₈ - NE₂₀₉ - NE₂₁₀ - NE₂₁₁ - NE₂₁₂ - NE₂₁₃ - NE₂₁₄ - NE₂₁₅ - NE₂₁₆ - NE₂₁₇ - NE₂₁₈ - NE₂₁₉ - NE₂₂₀ - NE₂₂₁ - NE₂₂₂ - NE₂₂₃ - NE₂₂₄ - NE₂₂₅ - NE₂₂₆ - NE₂₂₇ - NE₂₂₈ - NE₂₂₉ - NE₂₃₀ - NE₂₃₁ - NE₂₃₂ - NE₂₃₃ - NE₂₃₄ - NE₂₃₅ - NE₂₃₆ - NE₂₃₇ - NE₂₃₈ - NE₂₃₉ - NE₂₄₀ - NE₂₄₁ - NE₂₄₂ - NE₂₄₃ - NE₂₄₄ - NE₂₄₅ - NE₂₄₆ - NE₂₄₇ - NE₂₄₈ - NE₂₄₉ - NE₂₅₀ - NE₂₅₁ - NE₂₅₂ - NE₂₅₃ - NE₂₅₄ - NE₂₅₅ - NE₂₅₆ - NE₂₅₇ - NE₂₅₈ - NE₂₅₉ - NE₂₆₀ - NE₂₆₁ - NE₂₆₂ - NE₂₆₃ - NE₂₆₄ - NE₂₆₅ - NE₂₆₆ - NE₂₆₇ - NE₂₆₈ - NE₂₆₉ - NE₂₇₀ - NE₂₇₁ - NE₂₇₂ - NE₂₇₃ - NE₂₇₄ - NE₂₇₅ - NE₂₇₆ - NE₂₇₇ - NE₂₇₈ - NE₂₇₉ - NE₂₈₀ - NE₂₈₁ - NE₂₈₂ - NE₂₈₃ - NE₂₈₄ - NE₂₈₅ - NE₂₈₆ - NE₂₈₇ - NE₂₈₈ - NE₂₈₉ - NE₂₉₀ - NE₂₉₁ - NE₂₉₂ - NE₂₉₃ - NE₂₉₄ - NE₂₉₅ - NE₂₉₆ - NE₂₉₇ - NE₂₉₈ - NE₂₉₉ - NE₃₀₀ - NE₃₀₁ - NE₃₀₂ - NE₃₀₃ - NE₃₀₄ - NE₃₀₅ - NE₃₀₆ - NE₃₀₇ - NE₃₀₈ - NE₃₀₉ - NE₃₁₀ - NE₃₁₁ - NE₃₁₂ - NE₃₁₃ - NE₃₁₄ - NE₃₁₅ - NE₃₁₆ - NE₃₁₇ - NE₃₁₈ - NE₃₁₉ - NE₃₂₀ - NE₃₂₁ - NE₃₂₂ - NE₃₂₃ - NE₃₂₄ - NE₃₂₅ - NE₃₂₆ - NE₃₂₇ - NE₃₂₈ - NE₃₂₉ - NE₃₃₀ - NE₃₃₁ - NE₃₃₂ - NE₃₃₃ - NE₃₃₄ - NE₃₃₅ - NE₃₃₆ - NE₃₃₇ - NE₃₃₈ - NE₃₃₉ - NE₃₄₀ - NE₃₄₁ - NE₃₄₂ - NE₃₄₃ - NE₃₄₄ - NE₃₄₅ - NE₃₄₆ - NE₃₄₇ - NE₃₄₈ - NE₃₄₉ - NE₃₅₀ - NE₃₅₁ - NE₃₅₂ - NE₃₅₃ - NE₃₅₄ - NE₃₅₅ - NE₃₅₆ - NE₃₅₇ - NE₃₅₈ - NE₃₅₉ - NE₃₆₀ - NE₃₆₁ - NE₃₆₂ - NE₃₆₃ - NE₃₆₄ - NE₃₆₅ - NE₃₆₆ - NE₃₆₇ - NE₃₆₈ - NE₃₆₉ - NE₃₇₀ - NE₃₇₁ - NE₃₇₂ - NE₃₇₃ - NE₃₇₄ - NE₃₇₅ - NE₃₇₆ - NE₃₇₇ - NE₃₇₈ - NE₃₇₉ - NE₃₈₀ - NE₃₈₁ - NE₃₈₂ - NE₃₈₃ - NE₃₈₄ - NE₃₈₅ - NE₃₈₆ - NE₃₈₇ - NE₃₈₈ - NE₃₈₉ - NE₃₉₀ - NE₃₉₁ - NE₃₉₂ - NE₃₉₃ - NE₃₉₄ - NE₃₉₅ - NE₃₉₆ - NE₃₉₇ - NE₃₉₈ - NE₃₉₉ - NE₄₀₀ - NE₄₀₁ - NE₄₀₂ - NE₄₀₃ - NE₄₀₄ - NE₄₀₅ - NE₄₀₆ - NE₄₀₇ - NE₄₀₈ - NE₄₀₉ - NE₄₁₀ - NE₄₁₁ - NE₄₁₂ - NE₄₁₃ - NE₄₁₄ - NE₄₁₅ - NE₄₁₆ - NE₄₁₇ - NE₄₁₈ - NE₄₁₉ - NE₄₂₀ - NE₄₂₁ - NE₄₂₂ - NE₄₂₃ - NE₄₂₄ - NE₄₂₅ - NE₄₂₆ - NE₄₂₇ - NE₄₂₈ - NE₄₂₉ - NE₄₃₀ - NE₄₃₁ - NE₄₃₂ - NE₄₃₃ - NE₄₃₄ - NE₄₃₅ - NE₄₃₆ - NE₄₃₇ - NE₄₃₈ - NE₄₃₉ - NE₄₄₀ - NE₄₄₁ - NE₄₄₂ - NE₄₄₃ - NE₄₄₄ - NE₄₄₅ - NE₄₄₆ - NE₄₄₇ - NE₄₄₈ - NE₄₄₉ - NE₄₅₀ - NE₄₅₁ - NE₄₅₂ - NE₄₅₃ - NE₄₅₄ - NE₄₅₅ - NE₄₅₆ - NE₄₅₇ - NE₄₅₈ - NE₄₅₉ - NE₄₆₀ - NE₄₆₁ - NE₄₆₂ - NE₄₆₃ - NE₄₆₄ - NE₄₆₅ - NE₄₆₆ - NE₄₆₇ - NE₄₆₈ - NE₄₆₉ - NE₄₇₀ - NE₄₇₁ - NE₄₇₂ - NE₄₇₃ - NE₄₇₄ - NE₄₇₅ - NE₄₇₆ - NE₄₇₇ - NE₄₇₈ - NE₄₇₉ - NE₄₈₀ - NE₄₈₁ - NE₄₈₂ - NE₄₈₃ - NE₄₈₄ - NE₄₈₅ - NE₄₈₆ - NE₄₈₇ - NE₄₈₈ - NE₄₈₉ - NE₄₉₀ - NE₄₉₁ - NE₄₉₂ - NE₄₉₃ - NE₄₉₄ - NE₄₉₅ - NE₄₉₆ - NE₄₉₇ - NE₄₉₈ - NE₄₉₉ - NE₅₀₀ - NE₅₀₁ - NE₅₀₂ - NE₅₀₃ - NE₅₀₄ - NE₅₀₅ - NE₅₀₆ - NE₅₀₇ - NE₅₀₈ - NE₅₀₉ - NE₅₁₀ - NE₅₁₁ - NE₅₁₂ - NE₅₁₃ - NE₅₁₄ - NE₅₁₅ - NE₅₁₆ - NE₅₁₇ - NE₅₁₈ - NE₅₁₉ - NE₅₂₀ - NE₅₂₁ - NE₅₂₂ - NE₅₂₃ - NE₅₂₄ - NE₅₂₅ - NE₅₂₆ - NE₅₂₇ - NE₅₂₈ - NE₅₂₉ - NE₅₃₀ - NE₅₃₁ - NE₅₃₂ - NE₅₃₃ - NE₅₃₄ - NE₅₃₅ - NE₅₃₆ - NE₅₃₇ - NE₅₃₈ - NE₅₃₉ - NE₅₄₀ - NE₅₄₁ - NE₅₄₂ - NE₅₄₃ - NE₅₄₄ - NE₅₄₅ - NE₅₄₆ - NE₅₄₇ - NE₅₄₈ - NE₅₄₉ - NE₅₅₀ - NE₅₅₁ - NE₅₅₂ - NE₅₅₃ - NE₅₅₄ - NE₅₅₅ - NE₅₅₆ - NE₅₅₇ - NE₅₅₈ - NE₅₅₉ - NE₅₆₀ - NE₅₆₁ - NE₅₆₂ - NE₅₆₃ - NE₅₆₄ - NE₅₆₅ - NE₅₆₆ - NE₅₆₇ - NE₅₆₈ - NE₅₆₉ - NE₅₇₀ - NE₅₇₁ - NE₅₇₂ - NE₅₇₃ - NE₅₇₄ - NE₅₇₅ - NE₅₇₆ - NE₅₇₇ - NE₅₇₈ - NE₅₇₉ - NE₅₈₀ - NE₅₈₁ - NE₅₈₂ - NE₅₈₃ - NE₅₈₄ - NE₅₈₅ - NE₅₈₆ - NE₅₈₇ - NE₅₈₈ - NE₅₈₉ - NE₅₉₀ - NE₅₉₁ - NE₅₉₂ - NE₅₉₃ - NE₅₉₄ - NE₅₉₅ - NE₅₉₆ - NE₅₉₇ - NE₅₉₈ - NE₅₉₉ - NE₆₀₀ - NE₆₀₁ - NE₆₀₂ - NE₆₀₃ - NE₆₀₄ - NE₆₀₅ - NE₆₀₆ - NE₆₀₇ - NE₆₀₈ - NE₆₀₉ - NE₆₁₀ - NE₆₁₁ - NE₆₁₂ - NE₆₁₃ - NE₆₁₄ - NE₆₁₅ - NE₆₁₆ - NE₆₁₇ - NE₆₁₈ - NE₆₁₉ - NE₆₂₀ - NE₆₂₁ - NE₆₂₂ - NE₆₂₃ - NE₆₂₄ - NE₆₂₅ - NE₆₂₆ - NE₆₂₇ - NE₆₂₈ - NE₆₂₉ - NE₆₃₀ - NE₆₃₁ - NE₆₃₂ - NE₆₃₃ - NE₆₃₄ - NE₆₃₅ - NE₆₃₆ - NE₆₃₇ - NE₆₃₈ - NE₆₃₉ - NE₆₄₀ - NE₆₄₁ - NE₆₄₂ - NE₆₄₃ - NE₆₄₄ - NE₆₄₅ - NE₆₄₆ - NE₆₄₇ - NE₆₄₈ - NE₆₄₉ - NE₆₅₀ - NE₆₅₁ - NE₆₅₂ - NE₆₅₃ - NE₆₅₄ - NE₆₅₅ - NE₆₅₆ - NE₆₅₇ - NE₆₅₈ - NE₆₅₉ - NE₆₆₀ - NE₆₆₁ - NE₆₆₂ - NE₆₆₃ - NE₆₆₄ - NE₆₆₅ - NE₆₆₆ - NE₆₆₇ - NE₆₆₈ - NE₆₆₉ - NE₆₇₀ - NE₆₇₁ - NE₆₇₂ - NE₆₇₃ - NE₆₇₄ - NE₆₇₅ - NE₆₇₆ - NE₆₇₇ - NE₆₇₈ - NE₆₇₉ - NE₆₈₀ - NE₆₈₁ - NE₆₈₂ - NE₆₈₃ - NE₆₈₄ - NE₆₈₅ - NE₆₈₆ - NE₆₈₇ - NE₆₈₈ - NE₆₈₉ - NE₆₉₀ - NE₆₉₁ - NE₆₉₂ - NE₆₉₃ - NE₆₉₄ - NE₆₉₅ - NE₆₉₆ - NE₆₉₇ - NE₆₉₈ - NE₆₉₉ - NE₇₀₀ - NE₇₀₁ - NE₇₀₂ - NE₇₀₃ - NE₇₀₄ - NE₇₀₅ - NE₇₀₆ - NE₇₀₇ - NE₇₀₈ - NE₇₀₉ - NE₇₁₀ - NE₇₁₁ - NE₇₁₂ - NE₇₁₃ - NE₇₁₄ - NE₇₁₅ - NE₇₁₆ - NE₇₁₇ - NE₇₁₈ - NE₇₁₉ - NE₇₂₀ - NE₇₂₁ - NE₇₂₂ - NE₇₂₃ - NE₇₂₄ - NE₇₂₅ - NE₇₂₆ - NE₇₂₇ - NE₇₂₈ - NE₇₂₉ - NE₇₃₀ - NE₇₃₁ - NE₇₃₂ - NE₇₃₃ - NE₇₃₄ - NE₇₃₅ - NE₇₃₆ - NE₇₃₇ - NE₇₃₈ - NE₇₃₉ - NE₇₄₀ - NE₇₄₁ - NE₇₄₂ - NE₇₄₃ - NE₇₄₄ - NE₇₄₅ - NE₇₄₆ - NE₇₄₇ - NE₇₄₈ - NE₇₄₉ - NE₇₅₀ - NE₇₅₁ - NE₇₅₂ - NE₇₅₃ - NE₇₅₄ - NE₇₅₅ - NE₇₅₆ - NE₇₅₇ - NE₇₅₈ - NE₇₅₉ - NE₇₆₀ - NE₇₆₁ - NE₇₆₂ - NE₇₆₃ - NE₇₆₄ - NE₇₆₅ - NE₇₆₆ - NE₇₆₇ - NE₇₆₈ - NE₇₆₉ - NE₇₇₀ - NE₇₇₁ - NE₇₇₂ - NE₇₇₃ - NE₇₇₄ - NE₇₇₅ - NE₇₇₆ - NE₇₇₇ - NE₇₇₈ - NE₇₇₉ - NE₇₈₀ - NE₇₈₁ - NE₇₈₂ - NE₇₈₃ - NE₇₈₄ - NE₇₈₅ - NE₇₈₆ - NE₇₈₇ - NE₇₈₈ - NE₇₈₉ - NE₇₉₀ - NE₇₉₁ - NE₇₉₂ - NE₇₉₃ - NE₇₉₄ - NE₇₉₅ - NE₇₉₆ - NE₇₉₇ - NE₇₉₈ - NE₇₉₉ - NE₈₀₀ - NE₈₀₁ - NE₈₀₂ - NE₈₀₃ - NE₈₀₄ - NE₈₀₅ - NE₈₀₆ - NE₈₀₇ - NE₈₀₈ - NE₈₀₉ - NE₈₁₀ - NE₈₁₁ - NE₈₁₂ - NE₈₁₃ - NE₈₁₄ - NE₈₁₅ - NE₈₁₆ - NE₈₁₇ - NE₈₁₈ - NE₈₁₉ - NE₈₂₀ - NE₈₂₁ - NE₈₂₂ - NE₈₂₃ - NE₈₂₄ - NE₈₂₅ - NE₈₂₆ - NE₈₂₇ - NE₈₂₈ - NE₈₂₉ - NE₈₃₀ - NE₈₃₁ - NE₈₃₂ - NE₈₃₃ - NE₈₃₄ - NE₈₃₅ - NE₈₃₆ - NE₈₃₇ - NE₈₃₈ - NE₈₃₉ - NE₈₄₀ - NE₈₄₁ - NE₈₄₂ - NE₈₄₃ - NE₈₄₄ - NE₈₄₅ - NE₈₄₆ - NE₈₄₇ - NE₈₄₈ - NE₈₄₉ - NE₈₅₀ - NE₈₅₁ - NE₈₅₂ - NE₈₅₃ - NE₈₅₄ - NE₈₅₅ - NE₈₅₆ - NE₈₅₇ - NE₈₅₈ - NE₈₅₉ - NE₈₆₀ - NE₈₆₁ - NE₈₆₂ - NE₈₆₃ - NE₈₆₄ - NE₈₆₅ - NE₈₆₆ - NE₈₆₇ - NE₈₆₈ - NE₈₆₉ - NE₈₇₀ - NE₈₇₁ - NE₈₇₂ - NE₈₇₃ - NE₈₇₄ - NE₈₇₅ - NE₈₇₆ - NE₈₇₇ - NE₈₇₈ - NE₈₇₉ - NE₈₈₀ - NE₈₈₁ - NE₈₈₂ - NE₈₈₃ - NE₈₈₄ - NE₈₈₅ - NE₈₈₆ - NE₈₈₇ - NE₈₈₈ - NE₈₈₉ - NE₈₉₀ - NE₈₉₁ - NE₈₉₂ - NE₈₉₃ - NE₈₉₄ - NE₈₉₅ - NE₈₉₆ - NE₈₉₇ - NE₈₉₈ - NE₈₉₉ - NE₉₀₀ - NE₉₀₁ - NE₉₀₂ - NE₉₀₃ - NE₉₀₄ - NE₉₀₅ - NE₉₀₆ - NE₉₀₇ - NE₉₀₈ - NE₉₀₉ - NE₉₁₀ - NE₉₁₁ - NE₉₁₂ - NE₉₁₃ - NE₉₁₄ - NE₉₁₅ - NE₉₁₆ - NE₉₁₇ - NE₉₁₈ - NE₉₁₉ - NE₉₂₀ - NE₉₂₁ - NE₉₂₂ - NE₉₂₃ - NE₉₂₄ - NE₉₂₅ - NE₉₂₆ - NE₉₂₇ - NE₉₂₈ - NE₉₂₉ - NE₉₃₀ - NE₉₃₁ - NE₉₃₂ - NE₉₃₃ - NE₉₃₄ - NE₉₃₅ - NE₉₃₆ - NE₉₃₇ - NE₉₃₈ - NE₉₃₉ - NE₉₄₀ - NE₉₄₁ - NE₉₄₂ - NE₉₄₃ - NE₉₄₄ - NE₉₄₅ - NE₉₄₆ - NE₉₄₇ - NE₉₄₈ - NE₉₄₉ - NE₉₅₀ - NE₉₅₁ - NE₉₅₂ - NE₉₅₃ - NE₉₅₄ - NE₉₅₅ - NE₉₅₆ - NE₉₅₇ - NE₉₅₈ - NE₉₅₉ - NE₉₆₀ - NE₉₆₁ - NE₉₆₂ - NE₉₆₃ - NE₉₆₄ - NE₉₆₅ - NE₉₆₆ - NE₉₆₇ - NE₉₆₈ - NE₉₆₉ - NE₉₇₀ - NE₉₇₁ - NE₉₇₂ - NE₉₇₃ - NE₉₇₄ - NE₉₇₅ - NE₉₇₆ - NE₉₇₇ - NE₉₇₈ - NE₉₇₉ - NE₉₈₀ - NE₉₈₁ - NE₉₈₂ - NE₉₈₃ - NE₉₈₄ - NE₉₈₅ - NE₉₈₆ - NE₉₈₇ - NE₉₈₈ - NE₉₈₉ - NE₉₉₀ - NE₉₉₁ - NE₉₉₂ - NE₉₉₃ - NE₉₉₄ - NE₉₉₅ - NE₉₉₆ - NE₉₉₇ - NE₉₉₈ - NE₉₉₉ - NE₁₀₀₀ - NE₁₀₀₁ - NE₁₀₀₂ - NE₁₀₀₃ - NE₁₀₀₄ - NE₁₀₀₅ - NE₁₀₀₆ - NE₁₀₀₇ - NE₁₀₀₈ - NE₁₀₀₉ - NE₁₀₁₀ - NE₁₀₁₁ - NE₁₀₁₂ - NE₁₀₁₃ - NE₁₀₁₄ - NE₁₀₁₅ - NE₁₀₁₆ - NE₁₀₁₇ - NE₁₀₁₈ - NE₁₀₁₉ - NE₁₀₂₀ - NE₁₀₂₁ - NE₁₀₂₂ - NE₁₀₂₃ - NE₁₀₂₄ - NE₁₀₂₅ - NE₁₀₂₆ - NE₁₀₂₇ - NE₁₀₂₈ - NE₁₀₂₉ - NE₁₀₃₀ - NE₁₀₃₁ - NE₁₀₃₂ - NE₁₀₃₃ - NE₁₀₃₄ - NE₁₀₃₅ - NE₁₀₃₆ - NE₁₀₃₇ - NE₁₀₃₈ - NE₁₀₃₉ - NE₁₀₄₀ - NE₁₀₄₁ - NE₁₀₄₂ - NE₁₀₄₃ - NE₁₀₄₄ - NE₁₀₄₅ - NE₁₀₄₆ - NE₁₀₄₇ - NE₁₀₄₈ - NE₁₀₄₉ - NE₁₀₅₀ - NE₁₀₅₁ - NE₁₀₅₂ - NE₁₀₅₃ - NE₁₀₅₄ - NE₁₀₅₅ - NE₁₀₅₆ - NE₁₀₅₇ - NE₁₀₅₈ - NE₁₀₅₉ - NE₁₀₆₀ - NE₁₀₆₁ - NE₁₀₆₂ - NE₁₀₆₃ - NE₁₀₆₄ - NE₁₀₆₅ - NE₁₀₆₆ - NE₁₀₆₇ - NE₁

Conjectures

- ALL can be quarantined—in particular,
 $\text{QMA/qpoly} \subseteq \text{PostBQP/poly}$

Conjectures

- ALL can be quarantined—in particular,
 $\text{QMA/qpoly} \subseteq \text{PostBQP/poly}$

Conjectures

- ALL can be quarantined—in particular, $\text{QMA/qpoly} \subseteq \text{PostBQP/poly}$
- The question of whether classical advice can always replace quantum advice is not independent of ZF set theory

Conjectures

- ALL can be quarantined—in particular, $\text{QMA/qpoly} \subseteq \text{PostBQP/poly}$
- The question of whether classical advice can always replace quantum advice is not independent of ZF set theory
- Similarly for whether classical proofs can replace quantum proofs (i.e. $\text{QCMA} = \text{QMA}$)

Summary

- The state of n particles is an exponentially long vector. Welcome to Quantum World!

Summary

- The state of n particles is an exponentially long vector. Welcome to Quantum World!
- But for most complexity-theoretic purposes, it's no worse than a **probability distribution** being an exponentially long vector

No Signal

VGA-1