FerQ: Fermat Quotient Reformulation of High-Order Binary Optimisation

Phuong-Nam Nguyen[1], Seng W. Loke[1], Anil Prabhakar[2], Son N. Tran[1]
[1]School of Information Technology, Deakin University, Australia
[2]Department of Electrical Engineering, IIT Madras, India

Every degree-d Boolean monomial is a closed-form combination of $d-1$ Fermat quotients of one integer sum: **exact, with zero auxiliary variables**.
FerQ-Bc then emits an annealer-ready QUBO with $O(\log m)$ ancillas per chain.

Problem high-order energies, pairwise hardware

In machine learning
structured prediction · Markov random fields · neuro-symbolic reasoning · k-SAT, Max-k-SAT

In quantum computing
p-spin glasses · k-local Hamiltonians [72] · annealer hardware: 2-body couplers only (QUBO) [35]

HOBO: $E(\mathbf{x}) = \sum_{S\subseteq[n]} J_S \prod_{i\in S} x_i$
order $k = \max |S| \geq 3$, $\mathbf{x} \in \{0,1\}^n$

Quadratisation [57–61,36]: add a ancillas + penalty $\lambda \to$ QUBO on $n+a$ variables; search space 2^{n+a}, ancilla count traded against penalty size

FerQ (ours): an **exact, closed-form identity** — 0 ancillas, no penalties, same 2^n landscape; **FerQ-Bc** then emits a QUBO with $O(\log m)$ ancillas per chain

Key idea Fermat quotients carry every interaction

① **Fermat's little theorem**: $a^p \equiv a \pmod p$ for every integer a, so the **Fermat quotient** $\delta_p(a) = \dfrac{a - a^p}{p} \in \mathbb{Z}$ [70,71]

② **Lemma 3.1**: $\delta_p(A_S) = -\dfrac{1}{p}\sum_{q=2}^{\min(p,m)} N(p,q)\, e_q(S)$

$N(p,q) = q!\,S(p,q)$ counts surjections; $e_q(S) = \binom{A_S}{q}$, so every interaction among the m variables of S lives in one integer $A_S = \sum_{i\in S} x_i \in \{0,\dots,m\}$

③ **Theorem 3.2 (FerQ)**: for $|S| = d$ and the first $d-1$ primes p_k,

$$\prod_{i\in S} x_i = \sum_{k=1}^{d-1} \alpha_k^{(d)}\, \delta_{p_k}(A_S)$$

$\alpha^{(d)}$ = last row of $-M_d^{-1}$, $[M_d]_{k,q} = N(p_k, q+1)/p_k$

Applied to all monomials (Eq. 10, Alg. 1): exact $\tilde{E}$, same n variables.

Check: $d = 4$, primes 2, 3, 5

A_S	δ_2	δ_3	δ_5	$\sum_k \alpha_k \delta_{p_k}$
0	0	0	0	0
1	0	0	0	0
2	−1	−2	−6	0
3	−3	−8	−48	0
4	−6	−20	−204	**1**

$\alpha^{(4)} = (-1/2,\ 5/16,\ -1/48) \to 1[A_S{=}4]$

d	$\alpha^{(d)}$ (Table F.2)
2	-1
3	$1,\ -1/2$
4	$-1/2,\ 5/16,\ -1/48$
5	$1/5,\ -41/300,\ 1/72,\ -1/1800$

Where FerQ sits the role a quantum algorithm plays × the ML problem it serves

six surveys [1–6] + 50 representative works

rows: **role** of the quantum algorithm ↓ · columns: **ML problem** →

	Supervised learning classification · regression · kernels	**Unsupervised & generative** clustering · PCA · density models	**Structured reasoning** SAT · MRF-MAP · matching · graphs	**Agents & decisions** RL · multi-agent systems · LLM agents	**Scientific ML** PDEs · dynamics · simulation
Subroutine speed-up HHL · QPE · amplitude ampl. · quantum walks — fault-tolerant	QSVM [7]; HHL linear solvers [8]	qPCA [9]; Recommendation [10]; *caveat: dequantised [11]*	Backtracking / SAT search [12]	Quantum-enhanced agents [13]; RL with generative model [14]	Linear PDE solvers [15]; Nonlinear ODEs (Carleman) [16]
Parameterised model PQC / QNN · kernels · data re-uploading — NISQ · gate model	QNN classifiers [17]; Quantum kernels [18,19]; Quantum transformers [20]	qGAN [21]; Quantum autoencoder [22]	QNLP [23]; Quantum graph NNs [24]	VQC policies [25,26]; Quantum multi-agent RL [27]	Variational PDE solvers [28]; Differentiable circuits [29]
Optimiser annealing [55] · QAOA [56] Ising / QUBO / HOBO — annealer · gate model	QBoost classifier [30]; QUBO-trained SVM [31]; Feature selection [32,34]	Clustering (QAOA, QA) [33,34]	QUBO / HOBO encodings [35,36]; Vision: matching, MAP [37]; ★ **FerQ** NeurIPS 2026 auxiliary-free HOBO → QUBO	Coalition formation [38]	Annealer ODE solver [39]
Sampler Born machines · Boltzmann / Gibbs — gate model · annealer	Annealer DNN pre-training [40]	Born machines [41,42]; Quantum Boltzmann machine [43]	Ground-state sampling [44]	QBM-based RL [45]; Quantum-walk agents [46]	Quantum-enhanced MCMC [47]
Deployment where the quantum resource sits	**Single QPU / annealer** Minor-embedding [48] ★ FerQ-Bc		**Hybrid quantum–classical** VQA training loop [3]	**Distributed** Federated QML [49,50] · Circuit cutting [51] · DQC [52–54]	

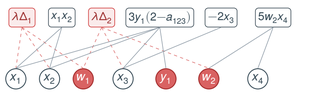
Worked example Fig. 1: $E = 5x_1x_2x_3x_4 - 3x_1x_2x_3 + x_1x_2 - 2x_3$

(a) Rosenberg substitution [57] — 6 vars · 2 ancillas



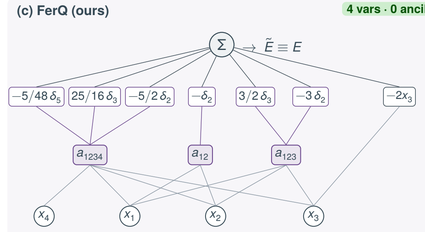
$w_1 = x_1x_2$, $w_2 = w_1x_3$, enforced by penalties $\lambda\Delta$

(b) HOCR [59] — 7 vars · 3 ancillas



exact gadget y_1 for $-3x_1x_2x_3$; w_1, w_2 for the quartic

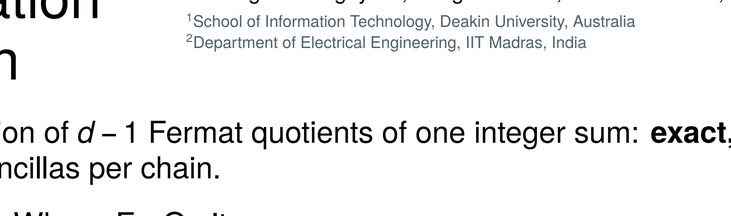
(c) FerQ (ours) — 4 vars · 0 ancillas



$a_S = \sum_{i\in S} x_i$ are linear sums, not new variables: the search space stays 2^4 and the landscape is preserved exactly (Eq. 4).

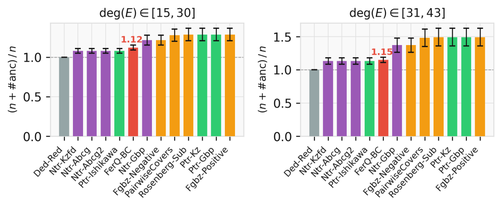
Results k-SAT and Max-k-SAT

SATLIB (24) and SATCOM'23–25 (30) for k-SAT; MSE2024 (26) and HAMLIB (30) for Max-k-SAT [66–69]. 12 baselines [36,57–65], incl. Rosenberg, HOCR (Ptr-Ishikawa), Ded-Red and FGBZ; one solver for all.



SATLIB: FerQ (red) has the highest median satisfaction and the lowest CPU runtime; the ranking holds on SATCOM'23–25, MSE2024 and HAMLIB. FerQ also matches the un-reformulated HOBO (0.989 vs 0.990 SAT).

Results qubit overhead of FerQ-Bc



1.15 qubits per variable at degree 31–43 (seven methods: 1.35–1.5)

150 vs 350–500 ancillas ($n = 1000$): 57–70% fewer than seven methods

degree ≤ 8: no gain (1.056, highest of 13) — use pairwise encodings there

$(n + \#\text{anc})/n$, 10,000 random energies, $n = 1000$ (Fig. B.2); Ded-Red (no ancillas, not universal), Ntr-* and Ptr-Ishikawa stay lower.

FerQ-Bc vs Rosenberg QUBOs (Tables F.5–F.6)

	SAT	MaxSAT	p-spin	QC-Hamiltonian
ancillas	−34% (10.5k vs 15.8k)	−53% (3.4k vs 7.2k)	−18% (81 vs 99)	−17% (82 vs 99)
quality	0.877 vs 0.875, n.s.	0.964 vs 0.958, n.s.	err. 0.613 vs **0.339**	err. **0.656** vs 1.553

FerQ-Bc an annealer-ready QUBO encoder

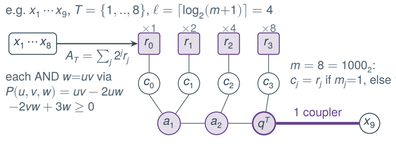
Lemma 3.3: a chain that tracks $A_T = \sum_{i\in T} x_i$ has an output qubit $q^T = 1[A_T{=}m] = \prod_{i\in T} x_i$.
Theorem 3.4: $\prod_{i\in S} x_i = x_{v_S} q^{T_S}$ with $T_S = S \setminus \{v_S\}$ → **one quadratic coupler per monomial**.

Domain wall, $m \leq 5$
e.g. $3x_1x_2x_3x_4x_5$, $T = \{1,..,4\}$

A_T	q_1	q_2	q_3	q_4	q^T
0	0	0	0	0	0
1	1	0	0	0	0
2	1	1	0	0	0
3	1	1	1	0	0
4	1	1	1	1	**1**

$q_1 \geq \dots \geq q_4$, $\sum_j q_j = A_T$ (penalties); $3x_1 \cdots x_5 = 3x_5q_4$, 4 ancillas

Binary counter, $m > 5$
e.g. $x_1 \cdots x_9$, $T = \{1,..,8\}$, $\ell = \lceil \log_2(m{+}1) \rceil = 4$



each AND $w{=}uv$ via $P(u,v,w) = uv - 2uw - 2vw + 3w \geq 0$

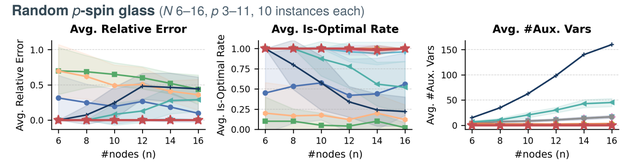
$m = 8 = 1000_2$: $c_j = r_j$ if $m_j{=}1$, else $1 - r_j$

ancillas $4 + 2 + 1 = 7 = \nu(8)$

Corollary 3.5: $\lambda > \sum |c_S|$ keeps the minimum; ancillas per chain $\nu(m) = \min(m,\ 2\lceil \log_2(m{+}1) \rceil - 1) = O(\log m)$; monomials sharing T reuse its chain (0 new ancillas).

Results exact ground states, zero ancillas

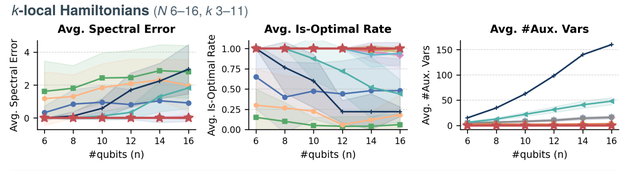
Random p-spin glass (N 6–16, p 3–11, 10 instances each)



99.6% optimal (23/24 configurations)

≈0 rel. error, 0 aux; ties the best ancilla-based methods ($p > 0.27$)

k-local Hamiltonians (N 6–16, k 3–11)



100% optimal, 0 spectral error, 0 aux

better than 5 methods ($p < 0.005$); ties Ntr-Gbp, Fgbz-Pos., Ptr-Kz

Summary take-home, limits and outlook

✓ Take-home
- any degree-d monomial = $d-1$ Fermat quotients of one integer sum
- exact, universal, closed form, 0 auxiliary variables
- lowest CPU runtime of 13 formulations
- FerQ-Bc: 1 coupler per monomial, best for degree $\gtrsim 8$

! Limits
- degree ≤ 43 with 64-bit coefficients (arbitrary precision lifts it)
- M_d inverted once per degree (offline)
- FerQ keeps the degree: annealers need FerQ-Bc
- low-degree, pairwise-dominant problems: pairwise encodings match or beat FerQ-Bc

→ Outlook
- co-design with next-generation annealers
- non-Boolean domains; continuous relaxations for gradient solvers
- ML uses: high-order MRFs (segmentation, denoising with FoE priors), structured labelling [63–65]

$\leq 2.6\times10^{-10}$ numerics
float64 vs exact rationals, any degree up to 43 (Table F.7)

5.4–12× GPU (CUDA, T4)
over CPU at 2,048 reads; slower than CPU at 50 reads

$\lambda \gtrsim \sum |c_S|$ suffices
for FerQ-Bc; 9–20-bit couplers for n = 50–1000 (Table F.8)

References [1] Biamonte+ Nature'17 [2] Dunjko & Briegel RPP'18 [3] Cerezo+ NRP'21 [4] Cerezo+ NCS'22 [5] Abbas+ NRP'24 [6] Marchisio+ arXiv'25 [7] Rebentrost+ PRL'14 [8] Harrow+ PRL'09 [9] Lloyd+ NatPhys'14 [10] Kerenidis & Prakash ITCS'17 [11] Tang STOC'19 [12] Montanaro ToC'18 [13] Dunjko+ PRL'16 [14] Wang+ ICML'21 [15] Childs+ Quantum'21 [16] Liu+ PNAS'21 [17] Farhi & Neven arXiv'18 [18] Havlíček+ Nature'19 [19] Liu+ NatPhys'21 [20] Khatri+ arXiv'24 [21] Zoufal+ npjQI'19 [22] Romero+ QST'17 [23] Lorenz+ JAIR'23 [24] Verdon+ arXiv'19 [25] Jerbi+ NeurIPS'21 [26] Skolik+ Quantum'22 [27] Yun+ ICDCS'22 [28] Lubasch+ PRA'20 [29] Kyriienko+ PRA'21 [30] Neven+ ACML'12 [31] Date+ SciRep'21 [32] Mücke+ QMI'23 [33] Otterbach+ arXiv'17 [34] Pomeroy+ arXiv'25 [35] Lucas FrontPhys'14 [36] Dattani arXiv'19 [37] Kuete Meli+ arXiv'25 [38] Mohseni+ QST'26 [39] Srivastava & Sundararaghavan PRA'19 [40] Adachi & Henderson arXiv'15 [41] Liu & Wang PRA'18 [42] Benedetti+ npjQI'19 [43] Amin+ PRX'18 [44] Mandrà+ PRL'17 [45] Crawford+ QIC'18 [46] Paparo+ PRX'14 [47] Layden+ Nature'23 [48] Choi QIP'08 [49] Chen & Yoo Entropy'21 [50] Nguyen+ arXiv'25 [51] Peng+ PRL'20 [52] Caleffi+ CompNet'24 [53] Barral+ CSR'25 [54] Knörzer+ arXiv'25 [55] Kadowaki & Nishimori PRE'98 [56] Farhi+ arXiv'14 [57] Rosenberg CahiersCERO'75 [58] Boros & Hammer DAM'02 [59] Ishikawa TPAMI'11 [60] Anthony+ MathProg'17 [61] Fix+ ICCV'11 [62] Tanburn+ arXiv'15 [63] Kolmogorov & Zabih TPAMI'04 [64] Freedman & Drineas CVPR'05 [65] Gallagher+ CVPR'11 [66] Hoos & Stützle SATLIB'00 [67] SAT Competition'23–25 [68] MaxSAT Evaluation'24 [69] Sawaya+ Quantum'24 [70] Moll Numbers & Functions'12 [71] Buium Invent.Math.'95 [72] Kempe+ SICOMP'06.

Supported by the Deakin–IIT Madras Joint Project (IITM-250601-157) and Deakin University (GR06084).

Deakin University CRICOS Provider Code: 00113B