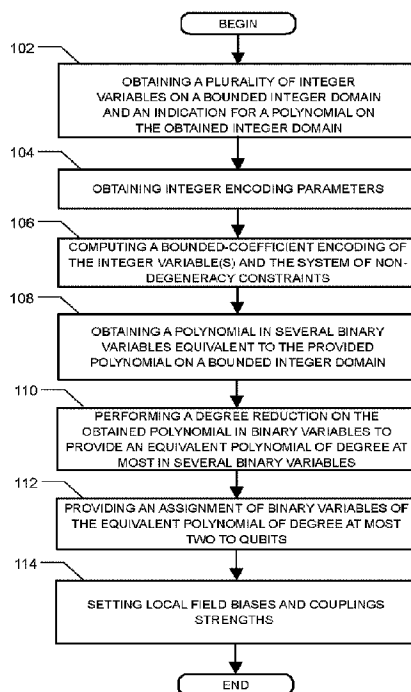




(86) Date de dépôt PCT/PCT Filing Date: 2017/05/26
(87) Date publication PCT/PCT Publication Date: 2017/11/30
(45) Date de délivrance/Issue Date: 2023/10/24
(85) Entrée phase nationale/National Entry: 2018/11/14
(86) N° demande PCT/PCT Application No.: CA 2017/050637
(87) N° publication PCT/PCT Publication No.: 2017/201626
(30) Priorité/Priority: 2016/05/26 (US15/165,655)

(51) Cl.Int./Int.Cl. *G06N 10/60* (2022.01)
(72) Inventeurs/Inventors:
KARIMI, SAHAR, CA;
RONAGH, POOYA, CA
(73) Propriétaire/Owner:
1QB INFORMATION TECHNOLOGIES INC., CA
(74) Agent: GOWLING WLG (CANADA) LLP

(54) Titre : PROCÉDES ET SYSTÈMES DE CONFIGURATION D'UN SYSTÈME DE QUBITS SUPRACONDUCTEURS AYANT UN REPRESENTANT HAMILTONIEN D'UN POLYNÔME SUR UN DOMAINE ENTIER LIMITE
(54) Title: METHODS AND SYSTEMS FOR SETTING A SYSTEM OF SUPER CONDUCTING QUBITS HAVING A HAMILTONIAN REPRESENTATIVE OF A POLYNOMIAL ON A BOUNDED INTEGER DOMAIN



(57) **Abrégé/Abstract:**

Described herein are methods, systems, and media for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. A polynomial on the bounded integer domain and integer encoding parameters may be obtained. Next, the bounded-coefficient encoding may be computed. Next, each integer variable of the polynomial may be recast as a linear function of binary variables. Next, coefficients of an equivalent binary representation of the polynomial may be computed. Next, a degree reduction may be performed on the equivalent binary representation of the polynomial to generate an equivalent polynomial of a degree of at most two in binary variables. Next, local field biases and coupling strengths may be set on the system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property
Organization
International Bureau

(43) International Publication Date
30 November 2017 (30.11.2017)



(10) International Publication Number
WO 2017/201626 A1

(51) International Patent Classification:
G06N 99/00 (2010.01)

(21) International Application Number:
PCT/CA2017/050637

(22) International Filing Date:
26 May 2017 (26.05.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
15/165,655 26 May 2016 (26.05.2016) US

(71) Applicant: **IQB INFORMATION TECHNOLOGIES INC.** [CA/CA]; 458-550 Burrard Street, Vancouver, British Columbia V6C 2B5 (CA).

(72) Inventors: **KARIMI, Sahar**; 458-550 Burrard Street, Vancouver, British Columbia V6C 2B5 (CA). **RONAGH, Pooya**; 458-550 Burrard Street, Vancouver, British Columbia V6C 2B5 (CA).

(74) Agent: **SCHROEDER, Hans** et al.; Gowling WLG (Canada) LLP, 160 Elgin Street, Suite 2600, Ottawa, Ontario K1P 1C3 (CA).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC,

(54) Title: METHODS AND SYSTEMS FOR SETTING A SYSTEM OF SUPER CONDUCTING QUBITS HAVING A HAMILTONIAN REPRESENTATIVE OF A POLYNOMIAL ON A BOUNDED INTEGER DOMAIN

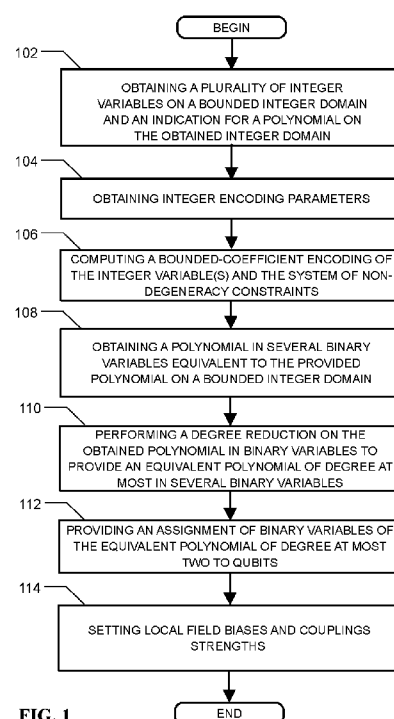


FIG. 1

(57) Abstract: Described herein are methods, systems, and media for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. A polynomial on the bounded integer domain and integer encoding parameters may be obtained. Next, the bounded-coefficient encoding may be computed. Next, each integer variable of the polynomial may be recast as a linear function of binary variables. Next, coefficients of an equivalent binary representation of the polynomial may be computed. Next, a degree reduction may be performed on the equivalent binary representation of the polynomial to generate an equivalent polynomial of a degree of at most two in binary variables. Next, local field biases and coupling strengths may be set on the system of superconducting qubits using the coefficients of the equivalent polynomial of the degree

[Continued on next page]



WO 2017/201626 A1

WO 2017/201626 A1

SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR,
TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

- (84) Designated States** (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

- *with international search report (Art. 21(3))*

METHODS AND SYSTEMS FOR SETTING A SYSTEM OF SUPER CONDUCTING QUBITS HAVING A HAMILTONIAN REPRESENTATIVE OF A POLYNOMIAL ON A BOUNDED INTEGER DOMAIN

[001]

BACKGROUND

[002] Quantum computers typically make use of quantum-mechanical phenomena, such as superposition and entanglement, to perform operations on data. Quantum computers may be different from digital electronic computers based on transistors. For instance, whereas digital computers require data to be encoded into binary digits (bits), each of which is always in one of two definite states (0 or 1), quantum computation uses quantum bits (qubits), which can be in superpositions of states.

[003] Systems of superconducting qubits are disclosed for instance in US Pat. Pub. Nos. US20120326720 and US20060225165, and manufactured by D-Wave Systems, IBM®, and Google®. Such analog systems may be used for implementing quantum computing algorithms, for example, the quantum adiabatic computation proposed by Farhi et. al. [arXiv:quant-ph/0001106] and Grover's quantum search algorithm [arXiv:quant-ph/0206003].

SUMMARY

[004] The present teachings relate to quantum information processing. Many methods exist for solving a binary polynomially constrained polynomial programming problem using a system of superconducting qubits. The methods disclosed herein can be used in conjunction with any method on any solver for solving a binary polynomially constrained polynomial programming problem to solve a mixed-integer polynomially constrained polynomial programming problem.

[005] Current implementations of quantum devices have limited numbers of superconducting qubits and are furthermore prone to various sources of noise. In practice, this restricts the usage of the quantum device to a limited number of qubits and a limited range of applicable ferromagnetic biases and couplings. Therefore, there is a need for methods of efficient encoding of data on the qubits of a quantum device.

[006] The present teachings relate to quantum information processing. This application pertains to a method for storing integers on superconducting qubits and setting a system of such superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain. Such a system of superconducting qubits may be configured to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient encoding.

[007] The methods disclosed herein can be used as a preprocessing operation for solving a mixed integer polynomially constrained polynomial programming problem with a solver for binary polynomially constrained polynomial programming problems. For example, this conversion may be achieved by casting each integer variable x as a linear function of binary variables, y_i for $i = 1, \dots, d$:

$$x = \sum_{i=1}^d c_i y_i$$

The tuple $(c_1, \dots, c_d)$ is referred to as an integer encoding. A few well-known integer encodings include:

[008] Binary Encoding, in which $c_i = 2^{i-1}$.

- Unary Encoding, in which $c_i = 1$.
- Sequential Encoding, in which $c_i = i$.

[009] Current implementations of quantum devices may have limited numbers of superconducting qubits and furthermore may be prone to various sources of noise, such as thermal and decoherence effects of the environment and the system [arXiv:1505.01545v2]. In practice, this may restrict the usage of the quantum device to a limited number of qubits and a limited range of applicable ferromagnetic biases and couplings.

[010] Consequently, the integer encodings formulated above may become incompetent for representing a polynomial in several integer variables as the Hamiltonian of the systems described above. The unary encoding may suffer from exploiting a large number of qubits. On the other hand, in the binary and sequential encodings, the coefficients c_i can be too large and therefore the behavior of the system may be affected considerably by the noise.

[011] In an aspect, disclosed herein is a method for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding, the method comprising: (a) using one or more computer processors to obtain (i) a polynomial on the bounded integer domain and (ii) integer encoding parameters; (b) computing the bounded-coefficient encoding using the integer encoding parameters; (c) recasting each integer variable of the polynomial as a linear function of binary variables using the bounded-coefficient encoding, and providing additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; (d) substituting each

integer variable of the polynomial with an equivalent binary representation, and computing coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; (e) performing a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and (f) setting local field biases and coupling strengths on the system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables.

[012] In some embodiments, the polynomial on the bounded integer domain is a single bounded integer variable. In some embodiments, (f) comprises assigning to a plurality of qubits a plurality of corresponding local field biases; wherein each local field bias corresponding to each of the qubits in the plurality of qubits is provided using the parameters of the integer encoding.

[013] In some embodiments, the polynomial on the bounded integer domain is a linear function of several bounded integer variables. In some embodiments, (f) comprises assigning to a plurality of qubits a plurality of corresponding local field biases; wherein each local field bias corresponding to each of the qubits in the plurality of qubits is provided using the linear function and the parameters of the integer encoding.

[014] In some embodiments, the polynomial on the bounded integer domain is a quadratic polynomial of several bounded integer variables. In some embodiments, (f) comprises embedding the equivalent binary representation of the polynomial of the degree of at most two on the bounded integer domain to a layout of the system of superconducting qubits comprising local fields on each of the plurality of the superconducting qubits and couplings in a plurality of pairs of the plurality of the superconducting qubits.

[015] In some embodiments, the system of superconducting qubits is a quantum annealer.

[016] In some embodiments, the method further comprises performing an optimization of the polynomial on the bounded integer domain via bounded-coefficient encoding. In some embodiments, the optimization of the polynomial on the bounded integer domain via bounded-coefficient encoding is performed by quantum adiabatic evolution of an initial transverse field on the superconducting qubits to a final Hamiltonian representative of the polynomial on the bounded integer domain on a measurable axis. In some embodiments, the optimization of the polynomial on the bounded integer domain via bounded-coefficient encoding comprises: (a) providing the equivalent polynomial of the degree of at most two in binary variables; (b) providing a system of non-degeneracy constraints; and (c) solving a problem of optimization of the equivalent polynomial of the degree of at most two in binary variables subject to the system

of non-degeneracy constraints as a binary polynomially constrained polynomial programming problem.

[017] In some embodiments, the method further comprises solving a polynomially constrained polynomial programming problem on a bounded integer domain via bounded-coefficient encoding. In some embodiments, solving the polynomially constrained polynomial programming problem on the bounded integer domain via bounded-coefficient encoding is performed by quantum adiabatic evolution of an initial transverse field on the superconducting qubits to a final Hamiltonian representative of the polynomial on the bounded integer domain on a measurable axis. In some embodiments, solving the polynomially constrained polynomial programming problem on the bounded integer domain via bounded-coefficient encoding comprises: (a) computing the bounded-coefficient encoding of an objective function and a set of constraints of the polynomially constrained polynomial programming problem using the integer encoding parameters to obtain an equivalent polynomially constrained polynomial programming problem in binary variables; (b) providing a system of non-degeneracy constraints; (c) adding the system of non-degeneracy constraints to a set of constraints of the equivalent polynomially constrained polynomial programming problem in binary variables; and (d) solving a problem of optimization of the equivalent polynomially constrained polynomial programming problem in binary variables.

[018] In some embodiments, the obtaining of the integer encoding parameters comprises obtaining an upper bound on coefficients of the bounded-coefficient encoding directly.

[019] In some embodiments, obtaining the integer encoding parameters comprises obtaining an upper bound on coefficients of the bounded-coefficient encoding based on error tolerances ϵ_f and ϵ_c of local field biases and coupling strengths, respectively, of the system of superconducting qubits. In some embodiments, obtaining the upper bound on the coefficients of the bounded-coefficient encoding comprises determining a feasible solution to a system of inequality constraints.

[020] In another aspect, disclosed herein is a system, comprising: (a) a sub-system of superconducting qubits; (b) a computer operatively coupled to the sub-system of superconducting qubits, wherein the computer comprises at least one computer processor, an operating system configured to perform executable instructions, and a memory; and (c) a computer program including instructions executable by the at least one computer processor to generate an application for setting the sub-system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-

coefficient encoding, the application comprising: i) a software module programmed or otherwise configured to obtain a polynomial on the bounded integer domain; ii) a software module programmed or otherwise configured to obtain integer encoding parameters; iii) a software module programmed or otherwise configured to compute the bounded-coefficient encoding using the integer encoding parameters; iv) a software module programmed or otherwise configured to (i) recast each integer variable of the polynomial to a linear function of binary variables using the bounded-coefficient encoding and (ii) provide additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; v) a software module programmed or otherwise configured to (i) substitute each integer variable of the polynomial with an equivalent binary representation and (ii) compute coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; vi) a software module programmed or otherwise configured to perform a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and vii) a software module programmed or otherwise configured to set local field biases and coupling strengths on the sub-system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables.

[021] In some embodiments, the polynomial on a bounded integer domain is a single bounded integer variable. In some embodiments, (c).vii) comprises assigning to a plurality of qubits a plurality of corresponding local field biases; wherein each local field bias corresponding to each of the qubits in the plurality of qubits is provided using the parameters of the integer encoding.

[022] In some embodiments, the polynomial on a bounded integer domain is a linear function of several bounded integer variables. In some embodiments, (c).vii) comprises assigning to a plurality of qubits a plurality of corresponding local field biases; wherein each local field bias corresponding to each of the qubits in the plurality of qubits is provided using the linear function and the parameters of the integer encoding.

[023] In some embodiments, the polynomial on a bounded integer domain is a quadratic polynomial of several bounded integer variables. In some embodiments, (c).vii) comprises embedding the equivalent binary representation of the polynomial of the degree of at most two on a bounded integer domain to a layout of the sub-system of superconducting qubits comprising local fields on each of the plurality of the superconducting qubits and couplings in a plurality of pairs of the plurality of the superconducting qubits.

[024] In some embodiments, the sub-system of superconducting qubits is a quantum annealer.

[025] In some embodiments, the system further comprises a software module programmed or otherwise configured to perform an optimization of the polynomial on the bounded integer domain via bounded-coefficient encoding.

[026] In some embodiments, the system further comprises a software module programmed or otherwise configured to solve a polynomially constrained polynomial programming problem on a bounded integer domain via bounded-coefficient encoding. In some embodiments, the obtaining of the integer encoding parameters comprises obtaining an upper bound on coefficients of the bounded-coefficient encoding directly. In some embodiments, obtaining the integer encoding parameters comprises obtaining an upper bound on coefficients of the bounded-coefficient encoding based on error tolerances ϵ_f and ϵ_c of local field biases and coupling strengths, respectively, of the sub-system of superconducting qubits.

[027] In another aspect, disclosed herein is a computer-readable medium comprising machine-executable code that, upon execution by one or more computer processors, implements a method for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding, the method comprising: (a) using the one or more computer processors to obtain (i) a polynomial of degree at most two on the bounded integer domain and (ii) integer encoding parameters; (b) computing the bounded-coefficient encoding using the integer encoding parameters; (c) recasting each integer variable of the polynomial as a linear function of binary variables using the bounded-coefficient encoding, and providing additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; (d) substituting each integer variable of the polynomial with an equivalent binary representation, and computing coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; (e) performing a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and (f) setting local field biases and coupling strengths on the system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables. In some embodiments, the computer-readable medium further comprises machine-executable code that, upon execution by the one or more computer processors, implements a method disclosed elsewhere herein.

[028] In an aspect, disclosed herein is a method for configuring a quantum computing system of superconducting qubits to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient encoding, the method comprising: (a) using one or more

computer processors to obtain (i) a polynomial on the bounded integer domain and (ii) integer encoding parameters; (b) computing the bounded-coefficient encoding using the integer encoding parameters; (c) transforming each integer variable of the polynomial to a linear function of binary variables using the bounded-coefficient encoding, and providing additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; (d) substituting each integer variable of the polynomial with an equivalent binary representation, and computing coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; (e) performing a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and (f) setting local field biases and coupling strengths on the quantum computing system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables to obtain a Hamiltonian representative of the polynomial on the bounded integer domain, which Hamiltonian is usable by the quantum computing system of superconducting qubits to solve the polynomial programming problem.

[029] In some embodiments, the polynomial on the bounded integer domain is a single bounded integer variable. In some embodiments, (f) comprises assigning to a plurality of qubits a plurality of corresponding local field biases; wherein each local field bias corresponding to each of the qubits in the plurality of qubits is provided using the parameters of the integer encoding.

[030] In some embodiments, the polynomial on the bounded integer domain is a linear function of several bounded integer variables. In some embodiments, (f) comprises assigning to a plurality of qubits a plurality of corresponding local field biases; wherein each local field bias corresponding to each of the qubits in the plurality of qubits is provided using the linear function and the parameters of the integer encoding.

[031] In some embodiments, the polynomial on the bounded integer domain is a quadratic polynomial of several bounded integer variables. In some embodiments, (f) comprises embedding the equivalent binary representation of the polynomial of the degree of at most two on the bounded integer domain to a layout of the quantum computing system of superconducting qubits comprising local fields on each of the plurality of the superconducting qubits and couplings in a plurality of pairs of the plurality of the superconducting qubits.

[032] In some embodiments, the system of superconducting qubits is a quantum annealer.

[033] In some embodiments, the method further comprises performing an optimization of the polynomial on the bounded integer domain via bounded-coefficient encoding. In some

embodiments, the optimization of the polynomial on the bounded integer domain via bounded-coefficient encoding is performed by quantum adiabatic evolution of an initial transverse field on the superconducting qubits to a final Hamiltonian representative of the polynomial on the bounded integer domain on a measurable axis. In some embodiments, the optimization of the polynomial on the bounded integer domain via bounded-coefficient encoding comprises: (a) providing the equivalent polynomial of the degree of at most two in binary variables; (b) providing a system of non-degeneracy constraints; and (c) solving a problem of optimization of the equivalent polynomial of the degree of at most two in binary variables subject to the system of non-degeneracy constraints as a binary polynomially constrained polynomial programming problem.

[034] In some embodiments, the method further comprises solving a polynomially constrained polynomial programming problem on a bounded integer domain via bounded-coefficient encoding. In some embodiments, solving the polynomially constrained polynomial programming problem on the bounded integer domain via bounded-coefficient encoding is performed by quantum adiabatic evolution of an initial transverse field on the superconducting qubits to a final Hamiltonian representative of the polynomial on the bounded integer domain on a measurable axis. In some embodiments, solving the polynomially constrained polynomial programming problem on the bounded integer domain via bounded-coefficient encoding comprises: (a) computing the bounded-coefficient encoding of an objective function and a set of constraints of the polynomially constrained polynomial programming problem using the integer encoding parameters to obtain an equivalent polynomially constrained polynomial programming problem in binary variables; (b) providing a system of non-degeneracy constraints; (c) adding the quantum computing system of non-degeneracy constraints to a set of constraints of the equivalent polynomially constrained polynomial programming problem in binary variables; and (d) solving a problem of optimization of the equivalent polynomially constrained polynomial programming problem in binary variables.

[035] In some embodiments, the obtaining of the integer encoding parameters comprises obtaining an upper bound on coefficients of the bounded-coefficient encoding directly.

[036] In some embodiments, obtaining the integer encoding parameters comprises obtaining an upper bound on coefficients of the bounded-coefficient encoding based on error tolerances ϵ_f and ϵ_c of local field biases and coupling strengths, respectively, of the quantum computing system of superconducting qubits. In some embodiments, obtaining the upper bound on the

coefficients of the bounded-coefficient encoding comprises determining a feasible solution to a system of inequality constraints.

[037] In another aspect, disclosed herein is a system for configuring a quantum computing subsystem of superconducting qubits to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient encoding, the system comprising: (a) the quantum computing subsystem of superconducting qubits; (b) a classical computer operatively coupled to the quantum computing subsystem of superconducting qubits, wherein the classical computer comprises at least one classical computer processor, an operating system configured to perform executable instructions, and a memory; and (c) a computer program including instructions executable by the at least one classical computer processor to generate an application for configuring the quantum computing subsystem of superconducting qubits to solve the polynomial programming problem on the bounded integer domain via bounded-coefficient encoding, the application comprising: i) a software module programmed or otherwise configured to obtain a polynomial on the bounded integer domain; ii) a software module programmed or otherwise configured to obtain integer encoding parameters; iii) a software module programmed or otherwise configured to compute the bounded-coefficient encoding using the integer encoding parameters; iv) a software module programmed or otherwise configured to (i) transform each integer variable of the polynomial to a linear function of binary variables using the bounded-coefficient encoding and (ii) provide additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; v) a software module programmed or otherwise configured to (i) substitute each integer variable of the polynomial with an equivalent binary representation and (ii) compute coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; vi) a software module programmed or otherwise configured to perform a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and vii) a software module programmed or otherwise configured to set local field biases and coupling strengths on the quantum computing subsystem of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables to obtain a Hamiltonian representative of the polynomial on the bounded integer domain, which Hamiltonian is usable by the quantum computing subsystem of superconducting qubits to solve the polynomial programming problem. In some embodiments, the method further comprises executing the quantum computing system of superconducting qubits having the Hamiltonian to solve the polynomial programming problem.

[038] In some embodiments, the polynomial on a bounded integer domain is a single bounded integer variable. In some embodiments, (c).vii) comprises assigning to a plurality of qubits a plurality of corresponding local field biases; wherein each local field bias corresponding to each of the qubits in the plurality of qubits is provided using the parameters of the integer encoding.

[039] In some embodiments, the polynomial on a bounded integer domain is a linear function of several bounded integer variables. In some embodiments, (c).vii) comprises assigning to a plurality of qubits a plurality of corresponding local field biases; wherein each local field bias corresponding to each of the qubits in the plurality of qubits is provided using the linear function and the parameters of the integer encoding.

[040] In some embodiments, the polynomial on a bounded integer domain is a quadratic polynomial of several bounded integer variables. In some embodiments, (c).vii) comprises embedding the equivalent binary representation of the polynomial of the degree of at most two on a bounded integer domain to a layout of the quantum computing subsystem of superconducting qubits comprising local fields on each of the plurality of the superconducting qubits and couplings in a plurality of pairs of the plurality of the superconducting qubits.

[041] In some embodiments, the quantum computing subsystem of superconducting qubits is a quantum annealer.

[042] In some embodiments, the system further comprises a software module programmed or otherwise configured to perform an optimization of the polynomial on the bounded integer domain via bounded-coefficient encoding.

[043] In some embodiments, the system further comprises a software module programmed or otherwise configured to solve a polynomially constrained polynomial programming problem on a bounded integer domain via bounded-coefficient encoding. In some embodiments, the obtaining of the integer encoding parameters comprises obtaining an upper bound on coefficients of the bounded-coefficient encoding directly. In some embodiments, obtaining the integer encoding parameters comprises obtaining an upper bound on coefficients of the bounded-coefficient encoding based on error tolerances ϵ_f and ϵ_c of local field biases and coupling strengths, respectively, of the quantum computing subsystem of superconducting qubits.

[044] In another aspect, disclosed herein is a computer-readable medium comprising machine-executable code that, upon execution by a classical computer, implements a method for configuring a quantum computing system of superconducting qubits to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient encoding, the

method comprising: (a) using one or more computer processors to obtain (i) a polynomial of degree at most two on the bounded integer domain and (ii) integer encoding parameters; (b) computing the bounded-coefficient encoding using the integer encoding parameters; (c) transforming each integer variable of the polynomial to a linear function of binary variables using the bounded-coefficient encoding, and providing additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; (d) substituting each integer variable of the polynomial with an equivalent binary representation, and computing coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; (e) performing a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and (f) setting local field biases and coupling strengths on the quantum computing system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables to obtain a Hamiltonian representative of the polynomial on the bounded integer domain, which Hamiltonian is usable by the quantum computing system of superconducting qubits to solve the polynomial programming problem. In some embodiments, the computer-readable medium further comprises machine-executable code that, upon execution by the one or more computer processors, implements a method disclosed elsewhere herein.

[045] In some embodiments, the obtaining of a polynomial in n variables on a bounded integer domain comprises providing the plurality of terms in the polynomial; each term of the polynomial further comprises the coefficient of the term and a list of size n representative of the power of each variables in the term in the matching index. The obtaining of a polynomial on a bounded integer domain further comprises obtaining a list of upper bounds on each integer variable.

[046] In a particular case where the provided polynomial has a degree of at most two, the obtaining of a polynomial on bounded domain comprises providing coefficients a_i of each linear term x_i for $i = 1, \dots, n$, and coefficients $Q_{ij} + Q_{ji}$ of each quadratic term $x_i x_j$ for all choices of distinct elements $\{i, j\} \subseteq \{1, \dots, n\}$ and an upper bound on each integer variable.

[047] In some embodiments, the obtaining of integer encoding parameters comprises either obtaining an upper bound on the value of the coefficients of the encoding directly; or obtaining the error tolerance ϵ_f and ϵ_c of the local field biases and couplings, respectively, and computing

the upper bound of the coefficients of the encoding from these error tolerances. This application proposes a technique for computing upper bound of the coefficients of the encoding from ϵ_f and ϵ_c for the special case that the provided polynomial has a degree of at most two.

[048] In some embodiments, the integer encoding parameters are obtained from at least one of a user, a computer, a software package and an intelligent agent.

[049] In some embodiments, the bounded-coefficient encoding is derived and the integer variables are represented as a linear function of a set of binary variables using the bounded-coefficient encoding, and a system of non-degeneracy constraints is returned.

[050] In another aspect, disclosed is a digital computer comprising: a central processing unit; a display device; a memory unit comprising an application for storing data and computing arithmetic operations; and a data bus for interconnecting the central processing unit, the display device, and the memory unit.

[051] In another aspect, there is disclosed a non-transitory computer-readable storage medium for storing computer-executable instructions which, when executed, cause a digital computer to perform arithmetic and logical operations.

[052] In another aspect, there is disclosed a transitory computer-readable signal medium for storing computer-executable instructions which, when executed, cause a digital computer to perform arithmetic and logical operations.

[053] In another aspect, there is disclosed a system of superconducting qubits comprising; a plurality of superconducting qubits; a plurality of couplings between a plurality of pairs of superconducting qubits; a quantum device control system capable of setting local field biases on each of the superconducting qubits and coupling strengths on each of the couplings.

[054] The methods disclosed herein makes it possible to represent a polynomial on a bounded integer domain on a system of superconducting qubits. The method comprises obtaining (i) the polynomial on the bounded integer domain and (ii) integer encoding parameters; computing the bounded-coefficient encoding using the integer encoding parameters; recasting each integer variable as a linear function of binary variables using the bounded-coefficient encoding, and providing additional constraints on the attained binary variables to avoid degeneracy in the encoding, if required by a user; substituting each integer variable with an equivalent binary representation, and computing the coefficients of the equivalent binary representation of the polynomial on the bounded integer domain; performing a degree reduction on the obtained equivalent binary representation of the polynomial on the bounded integer domain to provide an

equivalent polynomial of a degree of at most two in binary variables; and setting local field biases and coupling strengths on the system of superconducting qubits using the coefficients of the derived polynomial of a degree of at most two in several binary variables.

[055] In some embodiments, the methods disclosed herein makes it possible to find the optimal solution of a mixed integer polynomially constrained polynomial programming problem through solving its equivalent binary polynomially constrained polynomial programming problem. In some embodiments, solving a mixed integer polynomially constrained polynomial programming problem comprises finding a binary representation of all polynomials appearing the objective function and the constraints of the problem using the bounded-coefficient encoding and applying the methods proposed in US15/051271, US15/014576, CA2921711, and CA2881033 to the obtained equivalent binary polynomially constrained polynomial programming problem.

[056] Additional aspects and advantages of the present disclosure will become readily apparent to those skilled in this art from the following detailed description, wherein only illustrative embodiments of the present disclosure are shown and described. As will be realized, the present disclosure is capable of other and different embodiments, and its several details are capable of modifications in various obvious respects, all without departing from the disclosure. Accordingly, the drawings and description are to be regarded as illustrative in nature, and not as restrictive.

BRIEF DESCRIPTION OF THE DRAWINGS

[057] The novel features of the present teachings are set forth with particularity in the appended claims. A better understanding of the features and advantages of the present teachings will be obtained by reference to the following detailed description that sets forth illustrative embodiments, in which the principles of the present teachings are utilized, and the accompanying drawings (also “figure” and “FIG.” herein), of which:

[058] **FIG. 1** shows a non-limiting example of a method for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain; in this case, a flowchart of operations used for setting a system of superconducting qubits.

[059] **FIG. 2** shows a non-limiting example of a method for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain; in this case, a diagram of a system comprising of a digital computer interacting with a system of superconducting qubits.

[060] **FIG. 3** shows a non-limiting example of a method for setting a system of

superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain; in this case, a detailed diagram of a system comprising of a digital computer interacting with a system of superconducting qubits used for computing the local fields and couplers.

[061] FIG. 4 shows a non-limiting example of a method for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain; in this case, a flowchart of an operation for providing a polynomial on a bounded integer domain.

[062] FIG. 5 shows a non-limiting example of a method for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain; in this case, a flowchart of an operation for providing encoding parameters.

[063] FIG. 6 shows a non-limiting example of a method for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain; in this case, a flowchart of an operation for computing the bounded-coefficient encoding.

[064] FIG. 7 shows a non-limiting example of a method for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain; in this case, a flowchart of an operation for converting a polynomial on a bounded integer domain to an equivalent polynomial in several binary variables.

DETAILED DESCRIPTION

[065] While various embodiments of the present teachings have been shown and described herein, it will be obvious to those skilled in the art that such embodiments are provided by way of example only. Numerous variations, changes, and substitutions may occur to those skilled in the art without departing from the present teachings. It should be understood that various alternatives to the embodiments of the present teachings described herein may be employed.

[066] The methods disclosed herein can be applied to any quantum system of superconducting qubits, comprising local field biases on the qubits, and a plurality of couplings of the qubits, and control systems for applying and tuning local field biases and coupling strengths. Systems of quantum devices as such are disclosed for instance in US Pat. Pub. Nos. US 2012/0326720 and US 2006/0225165.

[067] The present teachings comprise a method for finding an integer encoding that uses the minimum number of binary variables in representation of an integer variable, while respecting an

upper bound on the values of coefficients appearing in the encoding. Such an encoding is referred to as a “bounded-coefficient encoding.” It also comprises a method for providing a system of constraints on the binary variables to prevent degeneracy of the bounded-coefficient encoding. Such a system of constraints involving the binary variables is referred to as “a system of non-degeneracy constraints.”

[068] The present teachings further comprise employing bounded-coefficient encoding to represent a polynomial on a bounded integer domain as the Hamiltonian of a system of superconducting qubits. Such a system of superconducting qubits may be configured to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient encoding.

[069] An advantage of the methods disclosed herein is that it enables an efficient method for finding the solution of a mixed integer polynomially constrained polynomial programming problem by finding the solution of an equivalent binary polynomially constrained polynomial programming. In some embodiments, the equivalent binary polynomially constrained polynomial programming problem may be solved by a system of superconducting qubits, for example, as disclosed in US15/051271, US15/014576, CA2921711, and CA2881033.

[070] Described herein is a method for configuring a quantum computing system of superconducting qubits to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient encoding, the method comprising: using one or more computer processors to obtain (i) a polynomial on the bounded integer domain and (ii) integer encoding parameters; computing the bounded-coefficient encoding using the integer encoding parameters; transforming each integer variable of the polynomial to a linear function of binary variables using the bounded-coefficient encoding, and providing additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; substituting each integer variable of the polynomial with an equivalent binary representation, and computing coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; performing a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and setting local field biases and coupling strengths on the quantum computing system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables to obtain a Hamiltonian representative of the polynomial on the bounded integer domain, which Hamiltonian is usable by the quantum computing system of superconducting qubits to solve the polynomial programming problem.

[071] Also described herein, in certain embodiments, is a system for configuring a quantum computing subsystem of superconducting qubits to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient encoding, the system comprising: the quantum computing subsystem of superconducting qubits; a classical computer operatively coupled to the quantum computing subsystem of superconducting qubits, wherein the classical computer comprises at least one classical computer processor, an operating system configured to perform executable instructions, and a memory; and a computer program including instructions executable by the at least one classical computer processor to generate an application for configuring the quantum computing subsystem of superconducting qubits to solve the polynomial programming problem on the bounded integer domain via bounded-coefficient encoding, the application comprising: a first software module programmed or otherwise configured to obtain a polynomial on the bounded integer domain; a second software module programmed or otherwise configured to obtain integer encoding parameters; a third software module programmed or otherwise configured to compute the bounded-coefficient encoding using the integer encoding parameters; a fourth software module programmed or otherwise configured to (i) transform each integer variable of the polynomial to a linear function of binary variables using the bounded-coefficient encoding and (ii) provide additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; a fifth software module programmed or otherwise configured to (i) substitute each integer variable of the polynomial with an equivalent binary representation and (ii) compute coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; a sixth software module programmed or otherwise configured to perform a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and a seventh software module programmed or otherwise configured to set local field biases and coupling strengths on the quantum computing subsystem of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables to obtain a Hamiltonian representative of the polynomial on the bounded integer domain, which Hamiltonian is usable by the quantum computing subsystem of superconducting qubits to solve the polynomial programming problem.

[072] Also described herein, in certain embodiments, is a computer-readable medium comprising machine-executable code that, upon execution by a classical computer, implements a method for configuring a quantum computing system of superconducting qubits to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient

encoding, the method comprising: using one or more computer processors to obtain (i) a polynomial of a degree of at most two on the bounded integer domain and (ii) integer encoding parameters; computing the bounded-coefficient encoding using the integer encoding parameters; transforming each integer variable of the polynomial to a linear function of binary variables using the bounded-coefficient encoding, and providing additional constraints on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user; substituting each integer variable of the polynomial with an equivalent binary representation, and computing coefficients of an equivalent binary representation of the polynomial on the bounded integer domain; performing a degree reduction on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables; and setting local field biases and coupling strengths on the quantum computing system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables to obtain a Hamiltonian representative of the polynomial on the bounded integer domain, which Hamiltonian is usable by the quantum computing system of superconducting qubits to solve the polynomial programming problem. The computer-readable medium may be non-transitory.

[073] The methods, systems, and media described herein may allow configuring a quantum computing system of superconducting qubits to produce higher quality solutions in response to a given computational task. Current quantum computer architectures may have limited numbers of superconducting qubits and consequently may be restricted in usage to a limited range of applicable ferromagnetic biases and couplings, thus limiting their utility to solving binary problems with binary variables. In practice, many discrete problems including polynomial programming problems, which may be expressed in terms of one or several integer variables, may necessitate a translation of the integer variables to binary variables in preparation for obtaining, on a quantum computing system of superconducting qubits, a Hamiltonian representative of the polynomial on the bounded integer domain. However, such a translation of integer variables to binary variables may represent a non-trivial task. Current transformation techniques may yield noisy solutions when solved on the quantum computing system of superconducting qubits. Since the methods, systems, and media described herein may lead to solutions to computational tasks that are of higher quality, fewer such computational tasks may be needed to be performed by a quantum computing system of superconducting qubits to converge to and obtain a final optimal solution to a given polynomial programming program. Similarly, more solutions may be obtained within a given time period as compared to other quantum computing approaches. Thus, quantum computing systems operating under the

disclosed methods, systems, and media may be significantly more efficient.

Definitions

[074] Unless otherwise defined, all technical terms used herein have the same meaning as commonly understood by one of ordinary skill in the art to which these present teachings belong. As used in this specification and the appended claims, the singular forms “a,” “an,” and “the” include plural references unless the context clearly dictates otherwise. Any reference to “or” herein is intended to encompass “and/or” unless otherwise stated.

[075] The term “integer variable” and like terms may refer to a data structure for storing integers in a digital system, between two integers ℓ and u where $\ell \leq u$. The integer ℓ may be called the “lower bound” and the integer u may be called the “upper bound” of the integer variable x .

[076] An integer variable $\bar{x}$ with lower and upper bounds ℓ and u , respectively, can be transformed to a bounded integer variable x with lower and upper bounds 0 and $u - \ell$, respectively.

[077] Accordingly, herein the term “bounded integer variable” may refer to an integer variable which may represent integer values with lower bound equal to 0. One may denote a bounded integer variable x with upper bound u by $x \in \{0, 1, \dots, u\}$.

[078] The term “binary variable” and like terms may refer to a data structure for storing integers 0 and 1 in a digital system. In some embodiments, computer bits are used to store such binary variables.

[079] The term “integer encoding” of a bounded integer variable x may refer to a tuple $(c_1, \dots, c_d)$ of integers such that the identity $\bar{x} = \sum_{i=1}^d c_i \bar{y}_i$ is satisfied for every possible value $\bar{x}$ of x using a choice of binary numbers $\bar{y}_1, \dots, \bar{y}_d$ for binary variables $y_1, \dots, y_d$.

[080] The term “bounded-coefficient encoding” with bound M , may refer to an integer encoding $(c_1, \dots, c_d)$ of a bounded integer variable x such that

$$c_i \leq M \text{ for all } i = 1, \dots, d$$

and may use the least number of binary variables $y_1, \dots, y_d$ amongst all encodings of x satisfying these inequalities.

[081] The term “a system of non-degeneracy constraints” may refer to a system of constraints that makes the equation $\bar{x} = \sum_{i=1}^d c_i \bar{y}_i$ have a unique binary solution $(\bar{y}_1, \dots, \bar{y}_d)$ for every choice of value $\bar{x}$ for variable x .

[082] The term “polynomial on a bounded integer domain” and like terms may refer to a function of the form

$$f(x) = \sum_{t=1}^T q_t \prod_{i=1}^n x_i^{p_i^t}.$$

in several integer variables $x_i \in \{0, 1, 2, \dots, \kappa_i\}$ for $i = 1, \dots, n$, where $p_i^t \geq 0$ is an integer denoting the power of variable x_i in t -th term and κ_i is the upper bound of x_i .

[083] The term “polynomial of a degree of at most two on bounded integer domain” and like terms may refer to a function of the form

$$f(x) = \sum_{i,j=1}^n Q_{ij} x_i x_j + \sum_{i=1}^n q_i x_i$$

in several integer variables $x_i \in \{0, 1, 2, \dots, \kappa_i\}$ for $i = 1, \dots, n$, where κ_i is the upper bound of x_i .

[084] A polynomial of degree at most two on binary domain, can be represented by a vector of linear coefficients $(q_1, \dots, q_n)$ and an $n \times n$ symmetric matrix $Q = (Q_{ij})$ with zero diagonal.

[085] The term “mixed-integer polynomially constrained polynomial programming” problem and like terms may refer to finding the minimum of a polynomial $y = f(x)$ in several variables $x = (x_1, \dots, x_n)$, such that a nonempty subset of them indexed by $S \subseteq \{1, \dots, n\}$ are bounded integer variables and the rest are binary variables, subject to a (possibly empty) family of equality constraints determined by a (possibly empty) family of e equations $g_j(x) = 0$ for $j = 1, \dots, e$ and a (possibly empty) family of inequality constraints determined by a (possibly empty) family of l inequalities $h_j(x) \leq 0$ for $j = 1, \dots, l$. Here, all functions $f(x)$, $g_i(x)$ for $i = 1, \dots, e$ and $h_j(x)$ for $j = 1, \dots, l$ may be polynomials. A mixed integer polynomially constrained polynomial programming problem can be represented as:

$$\begin{array}{ll}
\min & f(x) \\
\text{subject to} & g_i(x) = 0 \quad \forall i \in \{1, \dots, e\}, \\
& h_j(x) \leq 0 \quad \forall j \in \{1, \dots, l\}, \\
& x_s \in \{0, \dots, \kappa_s\} \quad \forall s \in S \subseteq \{1, \dots, n\}, \\
& x_s \in \{0, 1\} \quad \forall s \notin S.
\end{array}$$

The above mixed integer polynomially constrained polynomial programming problem may be denoted by (P_I) , and the optimal value of it may be denoted by $v(P_I)$. An optimal solution, denoted by x^* , may be a vector at which the objective function attains the value $v(P_I)$ and all constraints are satisfied.

[086] The term “polynomial of a degree of at most two on binary domain” and like terms may refer to a function of form $f(x) = \sum_{i,j=1}^n Q_{ij}x_i x_j + \sum_{i=1}^n q_i x_i$ defined on several binary variables $x_i \in \{0, 1\}$ for $i = 1, \dots, n$.

[087] A polynomial of a degree of at most two on binary domain, can be represented by a vector of linear coefficients $(q_1, \dots, q_n)$ and a $n \times n$ symmetric matrix $Q = (Q_{ij})$ with zero diagonal.

[088] The term “binary polynomially constrained polynomial programming” problem and like terms may refer to a mixed-integer polynomially constrained polynomial programming P_I such that $S = \emptyset$:

$$\begin{array}{ll}
\min & f(x) \\
\text{subject to} & g_i(x) = 0 \quad \forall i \in \{1, \dots, e\} \\
& h_j(x) \leq 0 \quad \forall j \in \{1, \dots, l\} \\
& x_k \in \{0, 1\} \quad \forall k \in \{1, \dots, n\}.
\end{array}$$

The above binary polynomially constrained polynomial programming problem may be denoted by P_B , and its optimal value may be denoted by $v(P_B)$.

[089] Two mathematical programming problems may be called “equivalent” if given the optimal solution of each one of them, the optimal solution of the other one can be computed in polynomial time of the size of the former optimal solution.

[090] The term “qubit” and like terms generally refer to any physical implementation of a quantum mechanical system represented on a Hilbert space and realizing at least two distinct and distinguishable eigenstates representative of the two states of a quantum bit. A quantum bit may

be an analog of a digital bit, where the ambient storing device may store two states $|0\rangle$ and $|1\rangle$ of a two-state quantum information, but also in superpositions

$$\alpha|0\rangle + \beta|1\rangle$$

of the two states. In various embodiments, such systems may have more than two eigenstates, in which case the additional eigenstates may be used to represent the two logical states by degenerate measurements. Various embodiments of implementations of qubits have been proposed; e.g. solid state nuclear spins, measured and controlled electronically or with nuclear magnetic resonance, trapped ions, atoms in optical cavities (cavity quantum-electrodynamics), liquid state nuclear spins, electronic charge or spin degrees of freedom in quantum dots, superconducting quantum circuits based on Josephson junctions (e.g., as described in Barone and Paternò, 1982, *Physics and Applications of the Josephson Effect*, John Wiley and Sons, New York; Martinis et al., 2002, *Physical Review Letters*, 89, 117901) and electrons on Helium.

[091] The term “local field,” may refer to a source of bias inductively coupled to a qubit. In some embodiments, a bias source is an electromagnetic device used to thread a magnetic flux through the qubit to provide control of the state of the qubit (e.g., as described in US Pat. Pub. No. US20060225165).

[092] The term “local field bias” and like terms may refer to a linear bias on the energies of the two states $|0\rangle$ and $|1\rangle$ of the qubit. In some embodiments, the local field bias is enforced by changing the strength of a local field in proximity of the qubit (e.g., as described in US Pat. Pub. No. US20060225165).

[093] The term “coupling” of two qubits H_1 and H_2 may refer to a device in proximity of both qubits threading a magnetic flux to both qubits. In some embodiments, a coupling may consist of a superconducting circuit interrupted by a compound Josephson junction. A magnetic flux may thread the compound Josephson junction and consequently thread a magnetic flux on both qubits (e.g., as described in US Pat. Pub. No. US20060225165).

[094] The term “coupling strength” between qubits H_1 and H_2 may refer to a quadratic bias on the energies of the quantum system comprising both qubits. In some embodiments, the coupling strength is enforced by tuning the coupling device in proximity of both qubits.

[095] The term “quantum device control system,” may refer to a system comprising a digital processing unit capable of initiating and tuning the local field biases and coupling strengths of a quantum system.

[096] The term “system of superconducting qubits” and like, may refer to a quantum mechanical system comprising a plurality of qubits and plurality of couplings between a plurality of pairs of the plurality of qubits. A system of superconducting qubits may further comprise a quantum device control system.

[097] A system of superconducting qubits may be manufactured in various embodiments. In some embodiments, a system of superconducting qubits is a “quantum annealer.”

[098] The term “quantum annealer” and like terms may refer to a system of superconducting qubits that carries optimization of a configuration of spins in an Ising spin model using quantum annealing as described, for example, in Farhi, E. et al., “Quantum Adiabatic Evolution Algorithms versus Simulated Annealing” arXiv.org: quant-ph/0201031 (2002), pp. 1-16. An embodiment of such an analog processor is disclosed by McGeoch, Catherine C. and Cong Wang, (2013), “Experimental Evaluation of an Adiabatic Quantum System for Combinatorial Optimization” Computing Frontiers,” May 14-16, 2013 and is also disclosed in US Pat. Pub. No. US20060225165.

Operations and architecture for setting a system of superconducting qubits

[099] In some embodiments, the methods, systems, and media described herein include a series of operations for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. In some embodiments, the methods disclosed herein can be used in conjunction with any method on any solver for solving a binary polynomially constrained polynomial programming problem to solve a mixed-integer polynomially constrained polynomial programming problem.

[0100] Referring to **FIG. 1**, in a particular embodiment, a flowchart of all operations is presented for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain. Specifically, processing operation 102 is shown to comprise obtaining a plurality of integer variables on a bounded integer domain and an indication for a polynomial in these variables. Processing operation 104 is disclosed to comprise obtaining integer encoding parameters. Processing operation 106 is used to comprise computing a bounded-coefficient encoding of the integer variable(s) and the system of non-degeneracy constraints. Processing operation 108 is displayed to comprise obtaining a polynomial in several binary variables equivalent to the provided polynomial on a bounded integer domain. Processing operation 110 is shown to comprise performing a degree reduction on the obtained polynomial in several binary variables to provide a polynomial of a degree of at most two in several binary variables. Processing operation 112 is shown to comprise providing an assignment of binary

variables of the equivalent polynomial of a degree of at most two to qubits. Processing operation 112 is shown to comprise setting local field biases and coupling strengths.

[0101] Referring to **FIG. 2**, in a particular embodiment, a diagram of a system for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain is demonstrated to comprise a digital computer interacting with a system of superconducting qubits.

[0102] Specially, there is shown an embodiment of a system 200 in which an embodiment of the method for setting a system of superconducting qubits in such a way that its Hamiltonian is representative of a polynomial on a bounded integer domain may be implemented. The system 200 comprises a digital computer 202 and a system 204 of superconducting qubits. The digital computer 202 receives a polynomial on a bounded integer domain and the encoding parameters and provides the bounded-coefficient encoding, a system of non-degeneracy constraints, and the values of local fields and couplers for the system of superconducting qubits.

[0103] The polynomial on a bounded integer domain may be provided according to various embodiments. In some embodiments, the polynomial on a bounded integer domain is provided by a user interacting with the digital computer 202. Alternatively, the polynomial on a bounded integer domain may be provided by another computer, not shown, operatively connected to the digital computer 202. Alternatively, the polynomial on a bounded integer domain may be provided by an independent software package. Alternatively, the polynomial on a bounded integer domain may be provided by an intelligent agent.

[0104] The integer encoding parameters may be provided according to various embodiments. In some embodiments, the integer encoding parameters are provided by a user interacting with the digital computer 202. Alternatively, the integer encoding parameters may be provided by another computer, not shown, operatively connected to the digital computer 202. Alternatively, the integer encoding parameters may be provided by an independent software package. Alternatively, the integer encoding parameters may be provided by an intelligent agent.

[0105] In some embodiments, the digital computer 202 may be any type. In some embodiments, the digital computer 202 is selected from a group consisting of desktop computers, laptop computers, tablet PCs, servers, smartphones, etc.

[0106] Referring to **FIG. 3**, in a particular embodiment, a diagram of a system for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain is demonstrated to comprise a digital computer used for computing the local fields and couplers.

[0107] Further referring to FIG. 3, there is shown an embodiment of a digital computer 202 interacting with a system 204 of superconducting qubits. The digital computer 202 may also be broadly referred to as a processor. In some embodiments, the digital computer 202 comprises a central processing unit (CPU) 302 (also referred to as a microprocessor), a display device 304, input devices 306, communication ports 308, a data bus 310, a memory unit 312, and a network interface card (NIC) 322.

[0108] The CPU 302 may be used for processing computer instructions. Various embodiments of the CPU 302 may be provided. In some embodiments, the central processing unit 302 is from Intel® and comprises a CPU Core i7-3820 running at 3.6 GHz.

[0109] The display device 304 may be used for displaying data to a user. Various types of display devices 304 may be used. In some embodiments, the display device 304 is a standard liquid crystal display (LCD) monitor.

[0110] The communication ports 308 may be used for sharing data with the digital computer 202. The communication ports 308 may comprise, for instance, a universal serial bus (USB) port for connecting a keyboard and a mouse to the digital computer 202. The communication ports 308 may further comprise a data network communication port such as an IEEE 802.3 port for enabling a connection of the digital computer 202 with another computer via a data network. Various alternative embodiments of the communication ports 308 may be provided. In some embodiments, the communication ports 308 comprise an Ethernet port and a mouse port (e.g., from Logitech®).

[0111] The memory unit 312 may be used for storing computer-executable instructions. The memory unit 312 may comprise an operating system module 314. The operating system module 314 may comprise one of various types. In an embodiment, the operating system module 314 is OS X Yosemite from Apple®.

[0112] The memory unit 312 may further comprise an application for providing a polynomial on a bounded integer domain, and integer encoding parameters 316. The memory unit 312 may further comprise an application for reducing the degree of a polynomial in several binary variables to a degree of at most two 318. The application for reducing the degree of a polynomial in several binary variables may comprise one of various kinds. An embodiment of an application for reducing a degree of a polynomial in several binary variables to a degree of at most two is disclosed in [H. Ishikawa, "Transformation of General Binary MRF Minimization to the First-Order Case," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 6, pp. 1234-1249, June 2011] and [Martin Anthony, Endre Boros, Yves Crama, and Arif Aritanan

Gruber, 2016, Quadraticization of symmetric pseudo-Boolean functions. Discrete Appl. Math. 203, C (April 2016), 1-12. DOI=<http://dx.doi.org/10.1016/j.dam.2016.01.001>]. The memory unit 312 may further comprise an application for minor embedding of a source graph to a target graph 320. The application for minor embedding may comprise one of various kinds. An embodiment of an application for minor embedding of a source graph to a target graph is disclosed in US Pat. No. US8244662. The memory unit 312 may further comprise an application for computing the local field biases and coupling strengths.

[0113] One or more of the central processing unit 302, the display device 304, the input devices 306, the communication ports 308, and the memory unit 312 may be interconnected via the data bus 310.

[0114] The system 202 may further comprise a network interface card (NIC) 322. The application 320 may send the appropriate signals along the data bus 310 into NIC 322. NIC 322, in turn, may send such information to quantum device control system 324.

[0115] The system 204 of superconducting qubits may comprise a plurality of superconducting quantum bits and a plurality of coupling devices. Further description of such a system is disclosed in US Pat. Pub. No. US20060225165.

[0116] The system 204 of superconducting qubits, may further comprise a quantum device control system 324. The control system 324 itself may comprise a coupling controller for each coupling in the plurality 328 of couplings of the device 204 capable of tuning the coupling strengths of a corresponding coupling, and local field bias controller for each qubit in the plurality 326 of qubits of the device 204 capable of setting a local field bias on each qubit.

Obtaining a plurality of integer variables on a bounded integer domain

[0117] In some embodiments, the methods, systems, and media described herein include a series of operations for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. In some embodiments, a processing operation is shown to comprise obtaining a plurality of integer variables on a bounded integer domain and an indication for a polynomial in those variables.

[0118] Referring to **FIG. 1** and according to processing operation 102, a polynomial on a bounded integer domain may be obtained. Referring to **FIG. 4**, in a particular embodiment, there is shown a detailed processing operation for providing a polynomial on a bounded integer domain.

[0119] According to processing operation 402, the coefficient of each term of a polynomial and

the degree of each variable in the corresponding term may be provided. Providing the coefficient and degree of each variable in each term can be performed in various embodiments. In some embodiments, a list of form $[Q_t, p_1^t, p_2^t, \dots, p_n^t]$ is provided for each term of the polynomial in which Q_t is the coefficient of the t -th term and p_i^t is the power of i -th variable in the t -th term.

[0120] In another embodiment, and in the particular case that the provided polynomial has a degree of at most two, a list $(q_1, \dots, q_n)$ and a $n \times n$ symmetric matrix $Q = (Q_{ij})$ is provided. A single bounded integer variable may be an embodiment of a polynomial of a degree of at most two in which $n = 1$, $q_1 = 1$ and $Q = (Q_{11}) = (0)$.

[0121] In some embodiments, if $Q_{ij} = 0$ for all $i, j = 1, \dots, n$, the provided polynomial is a linear function.

[0122] The providing of a polynomial may be performed according to various embodiments.

[0123] As mentioned above and in some embodiments, the coefficients of a polynomial are provided by a user interacting with the digital computer 202. Alternatively, the coefficients of a polynomial may be provided by another computer operatively connected to the digital computer 202. Alternatively, the coefficients of a polynomial may be provided by an independent software package. Alternatively, an intelligent agent may provide the coefficients of a polynomial.

[0124] According to processing operation 404, an upper bound on each bounded integer variable may be provided. Providing of upper bounds on the bounded integer variables may be performed according to various embodiments.

[0125] As mentioned above and in some embodiments, the upper bounds on the integer variables may be provided by a user interacting with the digital computer 202. Alternatively, the upper bounds on the integer variables may be provided by another computer operatively connected to the digital computer 202. Alternatively, the upper bounds on the integer variables may be provided by an independent software package or a computer readable and executable subroutine. Alternatively, an intelligent agent may provide the upper bounds on the integer variables.

Obtaining integer encoding parameters

[0126] In some embodiments, the methods, systems, and media described herein include a series of operations for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. In some embodiments, a processing operation is shown to comprise obtaining integer encoding parameters. Referring to **FIG. 1** and processing operation 104, the integer encoding parameters may be obtained.

[0127] The integer encoding parameters may comprise either obtaining an upper bound on the coefficients c_i 's of the bounded-coefficient encoding directly; or obtaining the error tolerances ϵ_f and ϵ_c of the local field biases and coupling strengths, respectively. If the upper bound on the coefficients c_i 's is not provided directly, it is computed by the digital computer 202 as described in the processing operation 504.

[0128] Referring to **FIG. 5** and according to processing operation 502, an upper bound on the coefficients of the bounded-coefficient encoding may be provided. The providing of the upper bound on the coefficients of the bounded-coefficient encoding may be performed according to various embodiments. In some embodiments, the upper bound on the coefficients of the bounded-coefficient encoding is provided directly by a user, a computer, a software package, or an intelligent agent.

[0129] Still referring to processing operation 502, if the upper bound on the coefficients of the bounded-coefficient encoding is not directly provided, the error tolerances of the local field biases and the coupling strengths of the system of superconducting qubits may be provided. The providing of the error tolerances of the local field biases and the coupling strengths of the system of superconducting qubits may be performed according to various embodiments. In some embodiments, the error tolerances of the local field biases and the coupling strengths of the system of superconducting qubits are provided directly by user, a computer, a software package, or an intelligent agent.

[0130] According to processing operations 504, the upper bound on the coefficients of the bounded-coefficient encoding is obtained based on the error tolerances ϵ_f and ϵ_c , respectively of the local field biases and coupling strengths of the system of superconducting qubits.

[0131] Still referring to processing operation 504, the upper bound of the values of the coefficients of the integer encoding may be obtained. The description of the system which may be used for computing the upper bound of the coefficients of the bounded-coefficient encoding

when ϵ_t and ϵ_c are provided, is now presented in detail.

[0132] If the provided polynomial is only a single bounded integer variable x , then the upper bound on the coefficients of the bounded-coefficient encoding of x denoted by μ^x may be computed and stored as $\mu^x = \left\lfloor \frac{1}{\epsilon_t} \right\rfloor$.

[0133] If the provided polynomial has a degree of one, i.e. $f(x) = \sum_{i=1}^n q_i x_i$, then the upper bound of the coefficients of the bounded-coefficient encoding for variable x_i may be computed and stored as $\mu^{x_i} = \left\lfloor \frac{\min_j \{|q_j|\}}{|q_i| \epsilon_t} \right\rfloor$.

[0134] If μ^{x_i} for $i = 1, \dots, n$ are required to be of equal value, the upper bound of the coefficients of the bounded-coefficient encoding may be computed and stored as

$$\mu = \left\lfloor \frac{\min_j \{|q_j|\}}{\max_j \{|q_j|\} \epsilon_t} \right\rfloor. \text{ This value of } \mu \text{ may coincide with } \min_j \{\mu^{x_j}\}.$$

[0135] If the provided polynomial comprises a degree of at least two, e.g.,

$f(x) = \sum_{t=1}^T Q_t \prod_{i=1}^n x_i^{p_i^t}$, and there exists a t such that $\sum_{i=1}^n p_i^t \geq 2$, the upper bounds on the coefficients of the bounded-coefficient encodings for variables x_i for $i = 1, \dots, n$ may be such that the coefficient of the equivalent polynomial with a degree of at most two in several variables derived after the substitution of binary representation of x_i 's and performing the degree

reduction, e.g., $f(x) = \sum_{i=1}^n q_i^B y_i + \sum_{i,j=1}^n Q_{ij}^B y_i y_j$ satisfy the following inequalities:

$$\frac{\min_i |q_i^B|}{\max_i |q_i^B|} \geq \epsilon_t,$$

and

$$\frac{\min_i |Q_{ij}^B|}{\max_i |Q_{ij}^B|} \geq \epsilon_c.$$

[0136] Finding the upper bounds on the coefficients of the bounded-coefficient encoding such that the above inequalities are satisfied can be done in various embodiments. In some embodiments, a variant of a bisection search is employed to find the upper bounds on the coefficients of the bounded-coefficient encoding such that the above inequalities are satisfied. In

another embodiment, a suitable heuristic search utilizing the coefficients and degree of the polynomial is employed to find the upper bounds on the coefficients of the bounded-coefficient encoding such that the above inequalities are satisfied.

[0137] In a particular case that $f(x) = \sum_{i=1}^n q_i x_i + \sum_{i,j=1}^n Q_{ij} x_i x_j$, and Q_{ii} and q_i are of the same sign, the above set of inequalities may be reduced to:

$$|Q_{ii}|(\mu^{x_i})^2 + |q_i|\mu^{x_i} \leq \frac{m_\ell}{\epsilon_\ell} \quad \text{for } i = 1, \dots, n,$$

$$\mu^{x_i} \leq \sqrt{\frac{m_c}{|Q_{ii}|\epsilon_c}} \quad \text{for } i = 1, \dots, n: Q_{ii} \neq 0,$$

$$\mu^{x_i} \mu^{x_j} \leq \frac{m_c}{|Q_{ij}|\epsilon_c} \quad \text{for } i, j = 1, \dots, n: Q_{ij} \neq 0,$$

for $m_\ell = \min_i \{|Q_{ii}| + q_i\}$ and $m_c = \min_{i,j} \{|Q_{ii}|, |Q_{ij}|\}$. Various methods may be employed to find μ^{x_i} for $i = 1, \dots, n$ that satisfy the above set of inequalities. In some embodiments, the following mathematical programming model may be solved with an appropriate solver on the digital computer 202 to find μ^{x_i} for $i = 1, \dots, n$.

$$\begin{aligned} \min \quad & \sum_{i=1}^n \frac{\kappa^{x_i}}{\mu^{x_i}}, \\ \text{subject to} \quad & Q_{ii}(\mu^{x_i})^2 + q_i \mu^{x_i} \leq \frac{m_\ell}{\epsilon_\ell} \quad \text{for } i = 1, \dots, n, \\ & \mu^{x_i} \leq \sqrt{\frac{m_c}{|Q_{ii}|\epsilon_c}} \quad \text{for } i = 1, \dots, n: Q_{ii} \neq 0, \\ & \mu^{x_i} \mu^{x_j} \leq \frac{m_c}{|Q_{ij}|\epsilon_c} \quad \text{for } i, j = 1, \dots, n: Q_{ij} \neq 0. \end{aligned}$$

In another embodiment, a heuristic search algorithm is employed for finding μ^{x_i} for $i = 1, 2, \dots, n$ that satisfy the above inequalities.

Computing a bounded-coefficient encoding of the integer variable(s) and the system of non-degeneracy constraints

[0138] In some embodiments, the methods, systems, and media described herein include a series of operations for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. In some embodiments, a processing operation is shown to comprise computing a bounded-coefficient encoding of the integer variable(s) and the system of non-degeneracy constraints. Referring to

FIG. 1 and processing operation 106, the bounded-coefficient encoding and the system of non-degeneracy constraints may be obtained.

[0139] Referring to **FIG. 6**, in a particular embodiment, described is how the bounded-coefficient encoding is derived. Herein, the upper bound on the integer variable x may be denoted with κ^x , and the upper bound on the coefficients used in the integer encoding may be denoted with μ^x . According to processing operation 602, the binary encoding of μ^x may be derived, setting $\ell_{\mu^x} = \lfloor \log_2 \mu^x + 1 \rfloor$. Then the binary encoding of μ may be set to:

$$S_{\mu^x} = (2^{i-1}; \text{for } i = 1, \dots, \ell_{\mu^x}).$$

If $\kappa^x < 2^{\ell_{\mu^x}}$ then the binary encoding of κ^x may not have any coefficients larger than μ^x ; hence, processing operation 602 derives

$$c_i^x = \begin{cases} 2^{i-1} & \text{for } i = 1, \dots, \lfloor \log_2 \kappa^x \rfloor \\ \kappa^x - \sum_{i=1}^{\lfloor \log_2 \kappa^x \rfloor} 2^{i-1} & \text{for } i = \lfloor \log_2 \kappa^x \rfloor + 1, \end{cases}$$

and processing operation 604 is skipped.

[0140] Still referring to **FIG. 6** and according to processing operation 604, the bounded-coefficient encoding may be completed, if required (e.g., $\kappa^x \geq 2^{\ell_{\mu^x}}$), by adding

$$\eta_{\mu^x} = \left\lfloor (\kappa^x - \sum_{i=0}^{\ell_{\mu^x}} 2^{i-1}) / \mu^x \right\rfloor \text{ coefficients of value } \mu, \text{ and one coefficient of value}$$

$\tau^x = \kappa^x - \sum_{i=0}^{\ell_{\mu^x}} 2^{i-1} - \eta_{\mu^x} \mu^x$ if τ is nonzero. Using the derived coefficients, the bounded-coefficient encoding may be the integer encoding in which the coefficients are as follows:

$$c_i^x = \begin{cases} 2^{i-1}, & \text{for } i = 1, \dots, \ell_{\mu^x}, \\ \mu^x, & \text{for } i = \ell_{\mu^x} + 1, \dots, \ell_{\mu^x} + \eta_{\mu^x}, \\ \tau^x, & \text{for } i = \ell_{\mu^x} + \eta_{\mu^x} + 1 \text{ if } \tau^x \neq 0. \end{cases}$$

[0141] The degree of the bounded-coefficient encoding may be

$$d^x = \begin{cases} \ell_{\mu^x} + \eta_{\mu^x} & \text{if } \tau^x = 0, \\ \ell_{\mu^x} + \eta_{\mu^x} + 1 & \text{otherwise.} \end{cases}$$

[0142] In the bounded-coefficient encoding, the following identity may be satisfied

$$\sum_{i=1}^{d^x} c_i^x = \kappa^x.$$

[0143] For example, if one needs to encode an integer variable that takes maximum value of 24 with integer encoding that has maximum coefficient of 6, the bounded-coefficient encoding may be

$$c_1 = 1, c_2 = 2, c_3 = 4, c_4 = 6, c_5 = 6, c_6 = 5.$$

[0144] The bounded-coefficient encoding may be derived according to various embodiment. In some embodiments, it is the output of a digital computer readable and executable subroutine.

[0145] Still referring to **FIG. 6** and according to processing operation 606, a system of non-degeneracy constraints may be provided. The system of non-degeneracy constraints may be represented in various embodiments.

[0146] In some embodiments, the system of non-degeneracy constraints may comprise the following system of linear inequalities:

$$\sum_{i=1}^{\ell_{\mu}^x} y_i^x \geq y_{\ell_{\mu}^x + 1}^x,$$

$$y_i^x \geq y_{i+1}^x, \quad \text{for } i = \ell_{\mu}^x + 1, \dots, d^x$$

[0147] The providing of the system of non-degeneracy constraints above may be carried by providing a matrix A of size $(d^x - \ell_{\mu}^x + 1) \times d^x$ with entries $-1, 0, 1$. In this embodiment, the system of non-degeneracy constraints is represented by the following system

$$A \begin{pmatrix} y_1^x \\ \vdots \\ y_{d^x}^x \end{pmatrix} \leq \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Converting from integer domain to binary variables

[0148] In some embodiments, the methods, systems, and media described herein include a series of operations for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. In some embodiments, a processing operation is shown to be providing a polynomial in several binary variables equivalent to the provide polynomial on a bounded integer domain. Referring back to **FIG. 1** and according to processing operation 108, the provided polynomial on a bounded integer domain may be converted to an equivalent polynomial in several binary variables.

[0149] Referring to **FIG. 7** and processing operation 702, each integer variable x_i may be

represented with the following linear function

$$x_i = \sum_{k=1}^{d^{x_i}} c_k^{x_i} y_k^{x_i}.$$

of binary variables $y_k^{x_i}$ for $k = 1, \dots, d^{x_i}$.

[0150] Still referring to **FIG. 7** and according to processing operation 704, the coefficients of the polynomial on binary variables equivalent to the obtained polynomial on bounded integer domain may be computed.

[0151] For each variable x_i in the obtained polynomial on a bounded integer domain, introduced herein are d^{x_i} binary variables $y_1^{x_i}, y_2^{x_i}, \dots, y_{d^{x_i}}^{x_i}$.

[0152] The coefficients of the polynomial in several binary variables may be computed in various embodiments.

[0153] In some embodiments, the computation of the coefficient of the polynomial in several binary variables may be performed according to methods disclosed in the documentation of the SymPy Python library for symbolic mathematics available online at [\[http://docs.sympy.org/latest/modules/polys/internals.html\]](http://docs.sympy.org/latest/modules/polys/internals.html) in conjunction to the relations of type $y^m = y$ for all binary variables.

[0154] In a particular case that the obtained polynomial on a bounded integer domain is linear, the resulting polynomial in binary variables is also linear and the coefficient of each variable $y_k^{x_i}$ may be expressed as $q_i c_k^{x_i}$ for $i = 1, \dots, n$ and $k = 1, \dots, d^{x_i}$.

[0155] In a particular case that the obtained polynomial on a bounded integer domain has a degree of two, then the equivalent polynomial in binary variables has a degree of two as well. Then, the coefficients of variable $y_k^{x_i}$ may be expressed as $q_i c_k^{x_i} + Q_{ii} (c_k^{x_i})^2$ for $i = 1, \dots, n$, and $k = 1, \dots, d^{x_i}$; the coefficients corresponding to $y_k^{x_i} y_l^{x_i}$ may be expressed as $Q_{ii} c_k^{x_i} c_l^{x_i}$ for $i = 1, \dots, n$, $k, l = 1, \dots, d^{x_i}$, and $k \neq l$; and the coefficients corresponding to $y_k^{x_i} y_l^{x_j}$ may be expressed as $Q_{ij} c_k^{x_i} c_l^{x_j}$ for $i, j = 1, \dots, n$, $i \neq j$, $k = 1, \dots, d^{x_i}$, and $l = 1, \dots, d^{x_j}$.

Degree reduction of the polynomial in several binary variables

[0156] In some embodiments, the methods, systems, and media described herein include a series

of operations for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. In some embodiments, a processing operation is shown to be providing a degree reduced form of a polynomial in several binary variables. Referring back to **FIG. 1** and according to processing operation 110, a polynomial having a degree of at most two in several binary variables is provided which is equivalent to the provided polynomial in several binary variables.

[0157] The degree reduction of a polynomial in several binary variables can be done in various embodiments. In some embodiments, the degree reduction of a polynomial in several binary variables is performed by the methods described in [H. Ishikawa, "Transformation of General Binary MRF Minimization to the First-Order Case," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 6, pp. 1234-1249, June 2011]. In another embodiment, the degree reduction of a polynomial in several binary variables is performed by the methods described in [Martin Anthony, Endre Boros, Yves Crama, and Aritanan Gruber, 2016, Quadraticization of symmetric pseudo-Boolean functions, Discrete Appl. Math, 203, C (April 2016), 1-12, DOI=<http://dx.doi.org/10.1016/j.dam.2016.01.001>].

Assigning variables to qubits

[0158] In some embodiments, the methods, systems, and media described herein include a series of operations for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. In some embodiments, a processing operation is shown to be providing an assignment of binary variables of the polynomial having a degree of at most two equivalent to the provided polynomial on bounded integer domain to qubits. Referring back to **FIG. 1** and according to processing operation 112, an assignment may be provided of the binary variables of the polynomial of a degree of at most two equivalent to the provided polynomial on bounded integer domain to qubits. In some embodiments, the assignment of binary variables to qubits is performed according to a minor embedding algorithm from a source graph obtained from the polynomial of a degree of at most two in several binary variables equivalent to the provided polynomial on bounded integer domain to a target graph obtained from the qubits and couplings of the pairs of qubits in the system of superconducting qubits.

[0159] A minor embedding from a source graph to a target graph may be performed according to various embodiments. In some embodiments, the algorithms disclosed in [A practical heuristic for finding graph minors – Jun Cai, Bill Macready, Aidan Roy] and/or in US Pat. Pub. No. US 20080218519 and US Pat. No. 8,655,828, are used.

Setting local field biases and coupling strengths

[0160] In some embodiments, the methods, systems, and media described herein include a series of operations for setting a system of superconducting qubits having a Hamiltonian representative of a polynomial on a bounded integer domain via bounded-coefficient encoding. In some embodiments, a processing operation is shown to be setting local field biases and coupling strengths. Referring back to **FIG. 1** and according to processing operation 114, the local field biases and coupling strengths on the system of superconducting qubits may be tuned.

[0161] In the particular case that the obtained polynomial is linear, each logical variable may be assigned a physical qubit and the local field bias of $q_i c_k^{x_i}$ may be assigned to the qubit corresponding to logical variable $y_k^{x_i}$ for $i = 1, \dots, n$ and $k = 1, \dots, d^{x_i}$.

[0162] In the particular case where the obtained polynomial has a degree of two or more, the degree reduced polynomial in several binary variables equivalent to the provided polynomial may be quadratic, and the tuning of local field biases and coupling strength may be carried according to various embodiments. In some embodiments, wherein the system of superconducting qubits is fully connected, each logical variable may be assigned a physical qubit. In this case, the local field of qubit corresponding to variable y may be set as the value of the coefficient of y in the polynomial having a degree of at most two in several binary variables. The coupling strength of the pair of qubits corresponding to variables y and y' may be set as the value of the coefficient of yy' in the polynomial having a degree of at most two in several binary variables.

[0163] The following example, illustrates how the method disclosed in this application may be used to recast a mixed-integer polynomially constrained polynomial programming problem to a binary polynomially constrained polynomial programming problem. Consider the optimization problem

$$\begin{aligned} \min \quad & (x_1 + x_3)^2 + x_2. \\ \text{subject to} \quad & x_1 + (x_2)^3 \leq 9, \\ & x_1, x_2 \in \mathbb{Z}_+, \\ & x_3 \in \{0,1\}. \end{aligned}$$

The above problem may be regarded as a mixed-integer polynomially constrained polynomial programming problem in which all the polynomials have a degree of at most three. According to

the constraint, an upper bound for the integer variable x_1 is 9 and an upper bound for the integer variable x_2 is 2.

[0164] Suppose one wants to convert this problem into an equivalent binary polynomially constrained polynomial programming with an integer encoding that has coefficients of at most three. The bounded-coefficient encoding for x_1 may be expressed as

$c_1^{x_1} = 1, c_2^{x_1} = 2, c_3^{x_1} = 3, c_4^{x_1} = 3$ and the bounded-coefficient encoding for x_2 may be expressed as $c_1^{x_2} = 1, c_2^{x_2} = 1$. The formal presentations for x_1 and x_2 may be expressed as

$$x_1 = y_1^{x_1} + 2y_2^{x_1} + 3y_3^{x_1} + 3y_4^{x_1},$$

$$x_2 = y_1^{x_2} + y_2^{x_2}.$$

[0165] Substituting the above linear functions for x_1 and x_2 in the mixed integer polynomially constrained polynomial programming problem, the following equivalent binary polynomially constrained polynomial programming problem may be obtained:

$$\begin{aligned} \min \quad & (y_1^{x_1} + 2y_2^{x_1} + 3y_3^{x_1} + 3y_4^{x_1} + x_3)^2 + y_1^{x_2} + y_2^{x_2}, \\ \text{subject to} \quad & y_1^{x_1} + 2y_2^{x_1} + 3y_3^{x_1} + 3y_4^{x_1} + (y_1^{x_2} + y_2^{x_2})^3 \leq 9, \\ & x_3, y_1^{x_1}, y_2^{x_1}, y_3^{x_1}, y_4^{x_1}, y_1^{x_2}, y_2^{x_2} \in \{0,1\}. \end{aligned}$$

[0166] If required, degenerate solutions may be ruled out by adding the system of non-degeneracy constraints provided by the methods disclosed herein, to the derived binary polynomially constrained polynomial programming problem as mentioned above. For the presented example, the final binary polynomially constrained polynomial programming problem may be expressed as:

$$\begin{aligned} \min \quad & (y_1^{x_1} + 2y_2^{x_1} + 3y_3^{x_1} + y_4^{x_1} + x_3)^2 + y_1^{x_2} + y_2^{x_2}, \\ \text{subject to} \quad & y_1^{x_1} + 2y_2^{x_1} + 3y_3^{x_1} + 3y_4^{x_1} + (y_1^{x_2} + y_2^{x_2})^3 \leq 9, \\ & y_1^{x_1} + y_2^{x_1} \geq y_3^{x_1}, \\ & y_3^{x_1} \geq y_4^{x_1}, \\ & y_1^{x_2} \geq y_2^{x_2}, \\ & x_3, y_1^{x_1}, y_2^{x_1}, y_3^{x_1}, y_4^{x_1}, y_1^{x_2}, y_2^{x_2} \in \{0,1\}. \end{aligned}$$

In this particular case, the first constraint of the above problem has a degree of three and is expressed in the form of

$$y_1^{x_1} + 2y_2^{x_1} + 3y_3^{x_1} + 3y_4^{x_1} + (y_1^{x_2})^3 + 3(y_1^{x_2})^2(y_2^{x_2}) + 3(y_1^{x_2})(y_2^{x_2})^2 + (y_2^{x_2})^3 \leq 9,$$

which can be equivalently represented as the degree reduced form of

$$y_1^{x_1} + 2y_2^{x_1} + 3y_3^{x_1} + 3y_4^{x_1} + (y_1^{x_2}) + 6(y_1^{x_2})(y_2^{x_2}) + (y_2^{x_2}) \leq 9.$$

Digital processing device

[0167] In some embodiments, the methods and systems described herein include a digital processing device, or use of the same. In further embodiments, the digital processing device includes one or more hardware central processing units (CPU) that carry out the device's functions. In still further embodiments, the digital processing device further comprises an operating system configured to perform executable instructions. In some embodiments, the digital processing device is optionally connected a computer network. In further embodiments, the digital processing device is optionally connected to the Internet such that it accesses the World Wide Web. In still further embodiments, the digital processing device is optionally connected to a cloud computing infrastructure. In other embodiments, the digital processing device is optionally connected to an intranet. In other embodiments, the digital processing device is optionally connected to a data storage device.

[0168] In accordance with the description herein, suitable digital processing devices include, by way of non-limiting examples, server computers, desktop computers, laptop computers, notebook computers, sub-notebook computers, netbook computers, netpad computers, set-top computers, media streaming devices, handheld computers, Internet appliances, mobile smartphones, tablet computers, personal digital assistants, video game consoles, and vehicles. Those of skill in the art will recognize that many smartphones are suitable for use in the system described herein. Those of skill in the art will also recognize that select televisions, video players, and digital music players with optional computer network connectivity are suitable for use in the system described herein. Suitable tablet computers include those with booklet, slate, and convertible configurations, known to those of skill in the art.

[0169] In some embodiments, the digital processing device includes an operating system configured to perform executable instructions. The operating system is, for example, software, including programs and data, which manages the device's hardware and provides services for execution of applications. Those of skill in the art will recognize that suitable server operating systems include, by way of non-limiting examples, FreeBSD, OpenBSD, NetBSD®, Linux, Apple® Mac OS X Server®, Oracle® Solaris®, Windows Server®, and Novell® NetWare®. Those of skill in the art will recognize that suitable personal computer operating systems include, by way of non-limiting examples, Microsoft® Windows®, Apple® Mac OS X®, UNIX®, and UNIX-like operating systems such as GNU/Linux®. In some embodiments, the operating system is provided by cloud computing. Those of skill in the art will also recognize that suitable mobile

smart phone operating systems include, by way of non-limiting examples, Nokia® Symbian® OS, Apple® iOS®, Research In Motion® BlackBerry OS®, Google® Android®, Microsoft® Windows Phone® OS, Microsoft® Windows Mobile® OS, Linux®, and Palm® WebOS®. Those of skill in the art will also recognize that suitable media streaming device operating systems include, by way of non-limiting examples, Apple TV®, Roku®, Boxee®, Google TV®, Google Chromecast®, Amazon Fire®, and Samsung® HomeSync®. Those of skill in the art will also recognize that suitable video game console operating systems include, by way of non-limiting examples, Sony® PS3®, Sony® PS4®, Microsoft® Xbox 360®, Microsoft Xbox One, Nintendo® Wii®, Nintendo® Wii U®, and Ouya®.

[0170] In some embodiments, the device includes a storage and/or memory device. The storage and/or memory device is one or more physical apparatuses used to store data or programs on a temporary or permanent basis. In some embodiments, the device is volatile memory and requires power to maintain stored information. In some embodiments, the device is non-volatile memory and retains stored information when the digital processing device is not powered. In further embodiments, the non-volatile memory comprises flash memory. In some embodiments, the non-volatile memory comprises dynamic random-access memory (DRAM). In some embodiments, the non-volatile memory comprises ferroelectric random access memory (FRAM). In some embodiments, the non-volatile memory comprises phase-change random access memory (PRAM). In other embodiments, the device is a storage device including, by way of non-limiting examples, CD-ROMs, DVDs, flash memory devices, magnetic disk drives, magnetic tapes drives, optical disk drives, and cloud computing based storage. In further embodiments, the storage and/or memory device is a combination of devices such as those disclosed herein.

[0171] In some embodiments, the digital processing device includes a display to send visual information to a user. In some embodiments, the display is a cathode ray tube (CRT). In some embodiments, the display is a liquid crystal display (LCD). In further embodiments, the display is a thin film transistor liquid crystal display (TFT-LCD). In some embodiments, the display is an organic light emitting diode (OLED) display. In various further embodiments, on OLED display is a passive-matrix OLED (PMOLED) or active-matrix OLED (AMOLED) display. In some embodiments, the display is a plasma display. In other embodiments, the display is a video projector. In still further embodiments, the display is a combination of devices such as those disclosed herein.

[0172] In some embodiments, the digital processing device includes an input device to receive information from a user. In some embodiments, the input device is a keyboard. In some embodiments, the input device is a pointing device including, by way of non-limiting examples,

a mouse, trackball, track pad, joystick, game controller, or stylus. In some embodiments, the input device is a touch screen or a multi-touch screen. In other embodiments, the input device is a microphone to capture voice or other sound input. In other embodiments, the input device is a video camera or other sensor to capture motion or visual input. In further embodiments, the input device is a Kinect, Leap Motion, or the like. In still further embodiments, the input device is a combination of devices such as those disclosed herein.

Computer readable medium

[0173] In some examples, a computer readable medium may comprise a non-transitory computer readable storage medium and/or a transitory computer readable signal medium. In some embodiments, the methods and systems disclosed herein include one or more non-transitory computer readable storage media and/or one or more transitory computer readable signal media encoded with a program including instructions executable by the operating system of an optionally networked digital processing device. In further embodiments, a computer readable storage medium is a tangible component of a digital processing device. In still further embodiments, a computer readable storage medium is optionally removable from a digital processing device. In some embodiments, a computer readable storage medium includes, by way of non-limiting examples, CD-ROMs, DVDs, flash memory devices, solid state memory, magnetic disk drives, magnetic tape drives, optical disk drives, cloud computing systems and services, and the like. In some cases, the program and instructions are permanently, substantially permanently, semi-permanently, or non-transitorily encoded on the media. In yet still further embodiments, a computer readable signal medium includes, by way of non-limiting examples, wireless signals such as RF, infrared or acoustic signals; or wire based signals such as electric impulses in a wire or optical impulses in a fiber optic cable.

Computer program

[0174] In some embodiments, the methods and systems disclosed herein include at least one computer program, or use of the same. A computer program may include a sequence of instructions, executable in the digital processing device's CPU, written to perform a specified task. Computer readable instructions may be implemented as program modules, such as functions, objects, Application Programming Interfaces (APIs), data structures, and the like, that perform particular tasks or implement particular abstract data types. In light of the disclosure provided herein, those of skill in the art will recognize that a computer program may be written in various versions of various languages.

[0175] The functionality of the computer readable instructions may be combined or distributed

as desired in various environments. In some embodiments, a computer program comprises one sequence of instructions. In some embodiments, a computer program comprises a plurality of sequences of instructions. In some embodiments, a computer program is provided from one location. In other embodiments, a computer program is provided from a plurality of locations. In various embodiments, a computer program includes one or more software modules. In various embodiments, a computer program includes, in part or in whole, one or more web applications, one or more mobile applications, one or more standalone applications, one or more web browser plug-ins, extensions, add-ins, or add-ons, or combinations thereof.

Web application

[0176] In some embodiments, a computer program includes a web application. In light of the disclosure provided herein, those of skill in the art will recognize that a web application, in various embodiments, utilizes one or more software frameworks and one or more database systems. In some embodiments, a web application is created upon a software framework such as Microsoft® .NET or Ruby on Rails (RoR). In some embodiments, a web application utilizes one or more database systems including, by way of non-limiting examples, relational, non-relational, object oriented, associative, and XML database systems. In further embodiments, suitable relational database systems include, by way of non-limiting examples, Microsoft® SQL Server, MySQL™, and Oracle®. Those of skill in the art will also recognize that a web application, in various embodiments, is written in one or more versions of one or more languages. A web application may be written in one or more markup languages, presentation definition languages, client-side scripting languages, server-side coding languages, database query languages, or combinations thereof. In some embodiments, a web application is written to some extent in a markup language such as Hypertext Markup Language (HTML), Extensible Hypertext Markup Language (XHTML), or eXtensible Markup Language (XML). In some embodiments, a web application is written to some extent in a presentation definition language such as Cascading Style Sheets (CSS). In some embodiments, a web application is written to some extent in a client-side scripting language such as Asynchronous Javascript and XML (AJAX), Flash® Actionscript, Javascript, or Silverlight®. In some embodiments, a web application is written to some extent in a server-side coding language such as Active Server Pages (ASP), ColdFusion®, Perl, Java™, JavaServer Pages (JSP), Hypertext Preprocessor (PHP), Python™, Ruby, Tcl, Smalltalk, WebDNA®, or Groovy. In some embodiments, a web application is written to some extent in a database query language such as Structured Query Language (SQL). In some embodiments, a web application integrates enterprise server products such as IBM® Lotus Domino®. In some embodiments, a web application includes a media player element. In various

further embodiments, a media player element utilizes one or more of many suitable multimedia technologies including, by way of non-limiting examples, Adobe® Flash®, HTML 5, Apple® QuickTime®, Microsoft® Silverlight®, Java™, and Unity®.

Mobile application

[0177] In some embodiments, a computer program includes a mobile application provided to a mobile digital processing device. In some embodiments, the mobile application is provided to a mobile digital processing device at the time it is manufactured. In other embodiments, the mobile application is provided to a mobile digital processing device via the computer network described herein.

[0178] In view of the disclosure provided herein, a mobile application is created by techniques known to those of skill in the art using hardware, languages, and development environments known to the art. Those of skill in the art will recognize that mobile applications are written in several languages. Suitable programming languages include, by way of non-limiting examples, C, C++, C#, Objective-C, Java™, Javascript, Pascal, Object Pascal, Python™, Ruby, VB.NET, WML, and XHTML/HTML with or without CSS, or combinations thereof.

[0179] Suitable mobile application development environments are available from several sources. Commercially available development environments include, by way of non-limiting examples, AirplaySDK, alcheMo, Appcelerator®, Celsius, Bedrock, Flash Lite, .NET Compact Framework, Rhomobile, and WorkLight Mobile Platform. Other development environments are available without cost including, by way of non-limiting examples, Lazarus, MobiFlex, MoSync, and Phonegap. Also, mobile device manufacturers distribute software developer kits including, by way of non-limiting examples, iPhone and iPad (iOS) SDK, Android™ SDK, BlackBerry® SDK, BREW SDK, Palm® OS SDK, Symbian SDK, webOS SDK, and Windows® Mobile SDK.

[0180] Those of skill in the art will recognize that several commercial forums are available for distribution of mobile applications including, by way of non-limiting examples, Apple® App Store, Android™ Market, BlackBerry® App World, App Store for Palm devices, App Catalog for webOS, Windows® Marketplace for Mobile, Ovi Store for Nokia® devices, Samsung® Apps, and Nintendo® DSi Shop.

Standalone application

[0181] In some embodiments, a computer program includes a standalone application, which is a program that is run as an independent computer process, not an add-on to an existing process, e.g., not a plug-in. Those of skill in the art will recognize that standalone applications are often compiled. A compiler is a computer program(s) that transforms source code written in a programming language into binary object code such as assembly language or machine code. Suitable compiled programming languages include, by way of non-limiting examples, C, C++, Objective-C, COBOL, Delphi, Eiffel, Java™, Lisp, Python™, Visual Basic, and VB .NET, or combinations thereof. Compilation is often performed, at least in part, to create an executable program. In some embodiments, a computer program includes one or more executable compiled applications.

Web browser plug-in

[0182] In some embodiments, the computer program includes a web browser plug-in. In computing, a plug-in is one or more software components that add specific functionality to a larger software application. Makers of software applications support plug-ins to enable third-party developers to create abilities which extend an application, to support easily adding new features, and to reduce the size of an application. When supported, plug-ins enable customizing the functionality of a software application. For example, plug-ins are commonly used in web browsers to play video, generate interactivity, scan for viruses, and display particular file types. Those of skill in the art will be familiar with several web browser plug-ins including, Adobe® Flash® Player, Microsoft® Silverlight®, and Apple® QuickTime®. In some embodiments, the toolbar comprises one or more web browser extensions, add-ins, or add-ons. In some embodiments, the toolbar comprises one or more explorer bars, tool bands, or desk bands.

[0183] In view of the disclosure provided herein, those of skill in the art will recognize that several plug-in frameworks are available that enable development of plug-ins in various programming languages, including, by way of non-limiting examples, C++, Delphi, Java™, PHP, Python™, and VB .NET, or combinations thereof.

[0184] Web browsers (also called Internet browsers) are software applications, designed for use with network-connected digital processing devices, for retrieving, presenting, and traversing information resources on the World Wide Web. Suitable web browsers include, by way of non-limiting examples, Microsoft® Internet Explorer®, Mozilla® Firefox®, Google® Chrome, Apple® Safari®, Opera Software® Opera®, and KDE Konqueror. In some embodiments, the web browser is a mobile web browser. Mobile web browsers (also called microbrowsers, mini-browsers, and

wireless browsers) are designed for use on mobile digital processing devices including, by way of non-limiting examples, handheld computers, tablet computers, netbook computers, subnotebook computers, smartphones, music players, personal digital assistants (PDAs), and handheld video game systems. Suitable mobile web browsers include, by way of non-limiting examples, Google® Android® browser, RIM BlackBerry® Browser, Apple® Safari®, Palm® Blazer, Palm® WebOS® Browser, Mozilla® Firefox® for mobile, Microsoft® Internet Explorer® Mobile, Amazon® Kindle® Basic Web, Nokia® Browser, Opera Software® Opera® Mobile, and Sony® PSP™ browser.

Software modules

[0185] In some embodiments, the methods and systems disclosed herein include software, server, and/or database modules, or use of the same. In view of the disclosure provided herein, software modules are created by techniques known to those of skill in the art using machines, software, and languages known to the art. The software modules disclosed herein are implemented in a multitude of ways. In various embodiments, a software module comprises a file, a section of code, a programming object, a programming structure, or combinations thereof. In further various embodiments, a software module comprises a plurality of files, a plurality of sections of code, a plurality of programming objects, a plurality of programming structures, or combinations thereof. In various embodiments, the one or more software modules comprise, by way of non-limiting examples, a web application, a mobile application, and a standalone application. In some embodiments, software modules are in one computer program or application. In other embodiments, software modules are in more than one computer program or application. In some embodiments, software modules are hosted on one machine. In other embodiments, software modules are hosted on more than one machine. In further embodiments, software modules are hosted on cloud computing platforms. In some embodiments, software modules are hosted on one or more machines in one location. In other embodiments, software modules are hosted on one or more machines in more than one location.

Databases

[0186] In some embodiments, the methods and systems disclosed herein include one or more databases, or use of the same. In view of the disclosure provided herein, those of skill in the art will recognize that many databases are suitable for storage and retrieval of application information. In various embodiments, suitable databases include, by way of non-limiting examples, relational databases, non-relational databases, object oriented databases, object databases, entity-relationship model databases, associative databases, and XML databases. In

some embodiments, a database is internet-based. In further embodiments, a database is web-based. In still further embodiments, a database is cloud computing-based. In other embodiments, a database is based on one or more local computer storage devices.

[0187] While preferred embodiments of the present teachings have been shown and described herein, it will be obvious to those skilled in the art that such embodiments are provided by way of example only. It is not intended that the teachings be limited by the specific examples provided within the specification. While the present teachings have been described with reference to the aforementioned specification, the descriptions and illustrations of the embodiments herein are not meant to be construed in a limiting sense. Numerous variations, changes, and substitutions will now occur to those skilled in the art without departing from the present teachings. Furthermore, it shall be understood that all aspects of the present teachings are not limited to the specific depictions, configurations or relative proportions set forth herein which depend upon a variety of conditions and variables. It should be understood that various alternatives to the embodiments of the present teachings described herein may be employed in practicing the present teachings. It is therefore contemplated that the present teachings shall also cover any such alternatives, modifications, variations or equivalents. It is intended that the following claims define the scope of the invention and that methods and structures within the scope of these claims and their equivalents be covered thereby.

[0188] Thus, from one perspective, there has now been described methods, systems, and media for configuring a quantum computing system of superconducting qubits to solve a polynomial programming problem on a bounded integer domain via bounded-coefficient encoding. One or more computer processors may be used to obtain a polynomial on the bounded integer domain and integer encoding parameters. Next, the bounded-coefficient encoding may be computed using the integer encoding parameters. Next, each integer variable of the polynomial may be transformed to a linear function of binary variables using the bounded-coefficient encoding, and additional constraints may be provided on the binary variables to avoid degeneracy in the bounded-coefficient encoding, if required by a user. Next, each integer variable of the polynomial may be substituted with an equivalent binary representation, and coefficients may be computed of an equivalent binary representation of the polynomial on the bounded integer domain. Next, a degree reduction may be performed on the equivalent binary representation of the polynomial on the bounded integer domain to generate an equivalent polynomial of a degree of at most two in binary variables. Next, local field biases and coupling strengths may be set on the quantum computing system of superconducting qubits using the coefficients of the equivalent polynomial of the degree of at most two in binary variables to obtain a Hamiltonian

representative of the polynomial on the bounded integer domain. The Hamiltonian may be usable by the quantum computing system of superconducting qubits to solve the polynomial programming problem.

CLAIMS

WHAT IS CLAIMED IS:

1. A method for using a digital computer to generate and direct a computational task to a quantum computing resource comprising at least one quantum computer over a network, wherein said digital computer comprises at least one computer processor and at least one computer memory, said method comprising: (a) retrieving a programming problem from said computer memory of said digital computer; (b) using said at least one computer processor of said digital computer to generate an equivalent of said programming problem; (c) generating a request comprising said equivalent of said programming problem generated by said at least one computer processor of said digital computer; and (d) directing said request from said digital computer to said quantum computing resource over said network, wherein said equivalent of said programming problem is usable by said at least one quantum computer of said quantum computing resource to solve said programming problem.
2. The method of claim 1, wherein said request is directed from said digital computer to said quantum computing resource through a cloud-based interface.
3. The method of claim 1, wherein said network is a local network.
4. The method of claim 1, wherein said at least one quantum computer performs one or more quantum algorithms to said programming problem.
5. The method of claim 1, wherein said request is generated using an application programming interface (API).
6. The method of claim 1, wherein (a) comprises obtaining (i) a polynomial on a bounded integer domain and (ii) integer encoding parameters, and wherein (b) comprises computing a bounded-coefficient encoding using said integer encoding parameters.
7. The method of claim 6, wherein (b) further comprises using said at least one computer processor to transform each integer variable of said polynomial to a linear function of binary variables using said bounded-coefficient encoding.

8. The method of claim 7, further comprising providing constraints on said binary variables to avoid degeneracy in said bounded-coefficient encoding, if required by a user.
9. The method of claim 7, further comprising substituting each integer variable of said polynomial with an equivalent binary representation, and using said at least one computer processor to compute coefficients of said equivalent binary representation of said polynomial on said bounded integer domain.
10. The method of claim 9, further comprising performing a degree reduction on said equivalent binary representation of said polynomial on said bounded integer domain to generate an equivalent polynomial.
11. The method of claim 10, wherein said equivalent polynomial is of a degree of at most two in binary variables.
12. The method of claim 11, further comprising setting local field biases and coupling strengths on said at least one quantum computer using said coefficients of said equivalent polynomial of said degree of at most two in binary variables to generate said equivalent of said programming problem.
13. The method of claim 12, wherein said equivalent of said programming problem comprises a Hamiltonian representative of said polynomial on said bounded integer domain.
14. The method of claim 13, wherein said Hamiltonian is usable by said at least one quantum computer to solve said programming problem.
15. A system comprising a digital computer for generating and directing a computational task to a quantum computing resource comprising at least one quantum computer over a network, wherein said digital computer comprises at least one computer processor and at least one computer memory, wherein said at least one computer processor is programmed to: (a) retrieve a programming problem from said computer memory of said digital computer; (b) use said at least one computer processor of said digital computer to generate an equivalent of said programming

problem; (c) generate a request comprising said equivalent of said programming problem generated by said at least one computer processor of said digital computer; and (d) direct said request from said digital computer to said quantum computing resource over said network, wherein said equivalent of said programming problem is usable by said at least one quantum computer of said quantum computing resource to solve said programming problem.

16. The system of claim 15, wherein said at least one computer processor is programmed to direct said request from said digital computer to said quantum computing resource through a cloud-based interface.

17. The system of claim 15, wherein said network is a local network.

18. The system of claim 15, wherein said at least one computer processor is programmed to obtain (i) a polynomial on a bounded integer domain and (ii) integer encoding parameters, and compute a bounded-coefficient encoding using said integer encoding parameters.

19. The system of claim 18, wherein said at least one computer processor is programmed to (i) transform each integer variable of said polynomial to a linear function of binary variables using said bounded-coefficient encoding, and (ii) substitute each integer variable of said polynomial with an equivalent binary representation, and compute coefficients of an equivalent binary representation of said polynomial on said bounded integer domain.

20. The system of claim 19, wherein said at least one computer processor is programmed to perform a degree reduction on said equivalent binary representation of said polynomial on said bounded integer domain to generate an equivalent polynomial, wherein said equivalent polynomial is of a degree of at most two in binary variables.

21. The system of claim 20, wherein said at least one computer processor is programmed to set local field biases and coupling strengths on said at least one quantum computer using said coefficients of said equivalent polynomial of said degree of at most two in binary variables to generate said equivalent of said programming problem.

22. The system of claim 21, wherein said equivalent of said programming problem comprises a Hamiltonian representative of said polynomial on said bounded integer domain.

23. The system of claim 22, wherein said Hamiltonian is usable by said at least one quantum computer to solve said programming problem.

24. A method for using a computer to generate a request comprising a computational task usable by at least one non-classical computer in communication with said computer over a distributed computing environment, wherein said computer comprises at least one computer processor, said method comprising: (a) using said at least one computer processor to (i) retrieve a first programming problem from computer memory, and (ii) using a bounded-coefficient encoding to transform said first programming problem from an integer representation to a binary representation to generate a second programming problem from said first programming problem; (b) generating said request comprising said computational task corresponding to said second programming problem in said computer memory, wherein said computational task is usable by said at least one non-classical computer to solve said second programming problem; and (c) directing said request from said computer to said non-classical computer over said distributed computing environment.

25. The method of claim 24, wherein said at least one non-classical computer is configured to perform one or more non-classical algorithms on said second programming problem.

26. The method of claim 24, wherein said request is generated using an application programming interface (API).

27. The method of claim 24, wherein said first programming problem comprises a polynomial on a bounded integer domain and integer encoding parameters, and wherein (a) further comprises computing said bounded-coefficient encoding using said integer encoding parameters to generate said second programming problem.

28. The method of claim 27, wherein (a) further comprises recasting each integer variable of said polynomial to a linear function of binary variables using said bounded-coefficient encoding to yield said second programming problem.

29. The method of claim 28, further comprising providing constraints on said binary variables for no degeneracy in said bounded-coefficient encoding.
30. The method of claim 27, wherein said second programming problem comprises a Hamiltonian representative of said polynomial on said bounded integer domain.
31. The method of claim 30, wherein said Hamiltonian is usable by said at least one non-classical computer to solve said second programming problem.
32. The method of claim 24, further comprising directing said request from said computer to said at least one non-classical computer over said distributed computing environment.
33. The method of claim 24, wherein said first programming problem comprises a mixed-integer polynomially constrained polynomial programming problem, and wherein said second programming problem comprises a binary polynomially constrained polynomial programming problem.
34. A system comprising a computer for generating a request comprising a computational task usable by at least one non-classical computer in communication with said computer over a distributed computing environment, wherein said computer comprises at least one computer processor, wherein said computer is configured to: (a) use said at least one computer processor to (i) retrieve a first programming problem from computer memory, and (ii) use a bounded-coefficient encoding to transform said first programming problem from an integer representation to a binary representation to generate a second programming problem from said first programming problem; (b) generate said request comprising said computational task corresponding to said second programming problem in said computer memory, wherein said computational task is usable by said at least one non-classical computer to solve said second programming problem; and (c) directing said request from said computer to said non-classical computer over said distributed computing environment.
35. The system of claim 34, wherein said computer is configured to generate said request using an application programming interface (API).

36. The system of claim 34, wherein said first programming problem in comprises a polynomial on a bounded integer domain and integer encoding parameters, and wherein in (b) said computer is configured to compute said bounded-coefficient encoding using said integer encoding parameters to generate said second programming problem.

37. The system of claim 36, wherein in (b) said computer is configured to recast each integer variable of said polynomial to a linear function of binary variables using said bounded-coefficient encoding to yield said second programming problem.

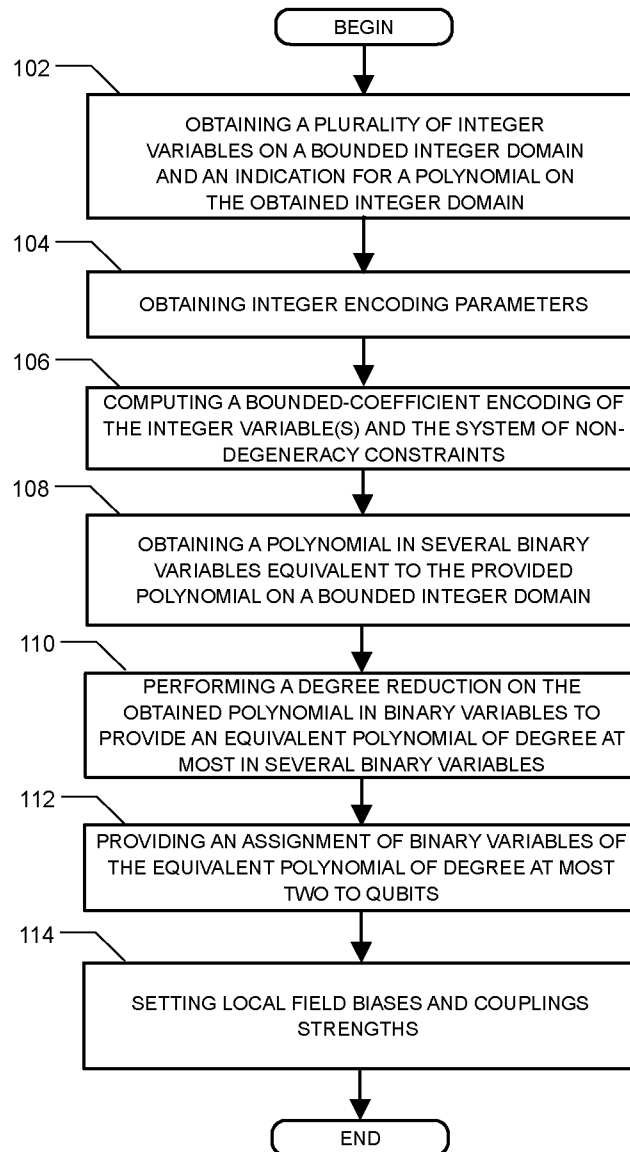
38. The system of claim 37, wherein said computer is configured to provide constraints on said binary variables for no degeneracy in said bounded-coefficient encoding.

39. The system of claim 36, wherein said second programming problem comprises a Hamiltonian representative of said polynomial on said bounded integer domain.

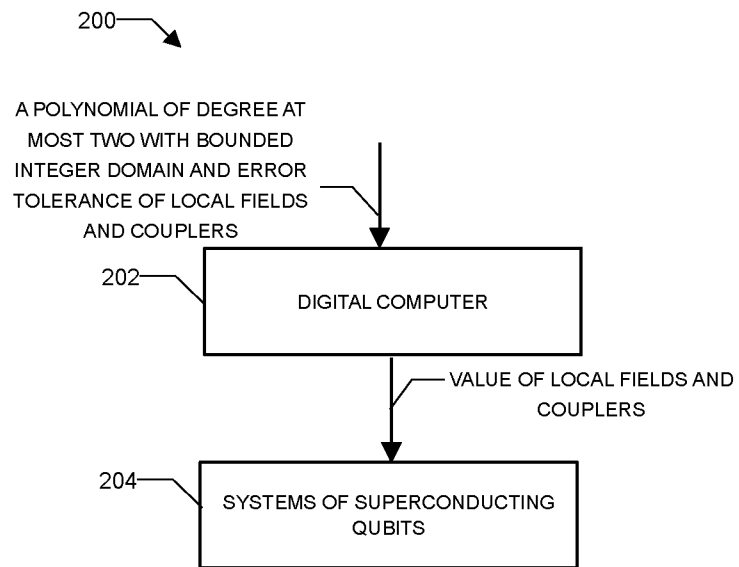
40. The system of claim 34, wherein said computer is configured to direct said request to said at least one non-classical computer over said distributed computing environment.

41. The system of claim 34, wherein said first programming problem comprises a mixed-integer polynomially constrained polynomial programming problem, and wherein said second programming problem comprises a binary polynomially constrained polynomial programming problem.

1/7
FIG. 1

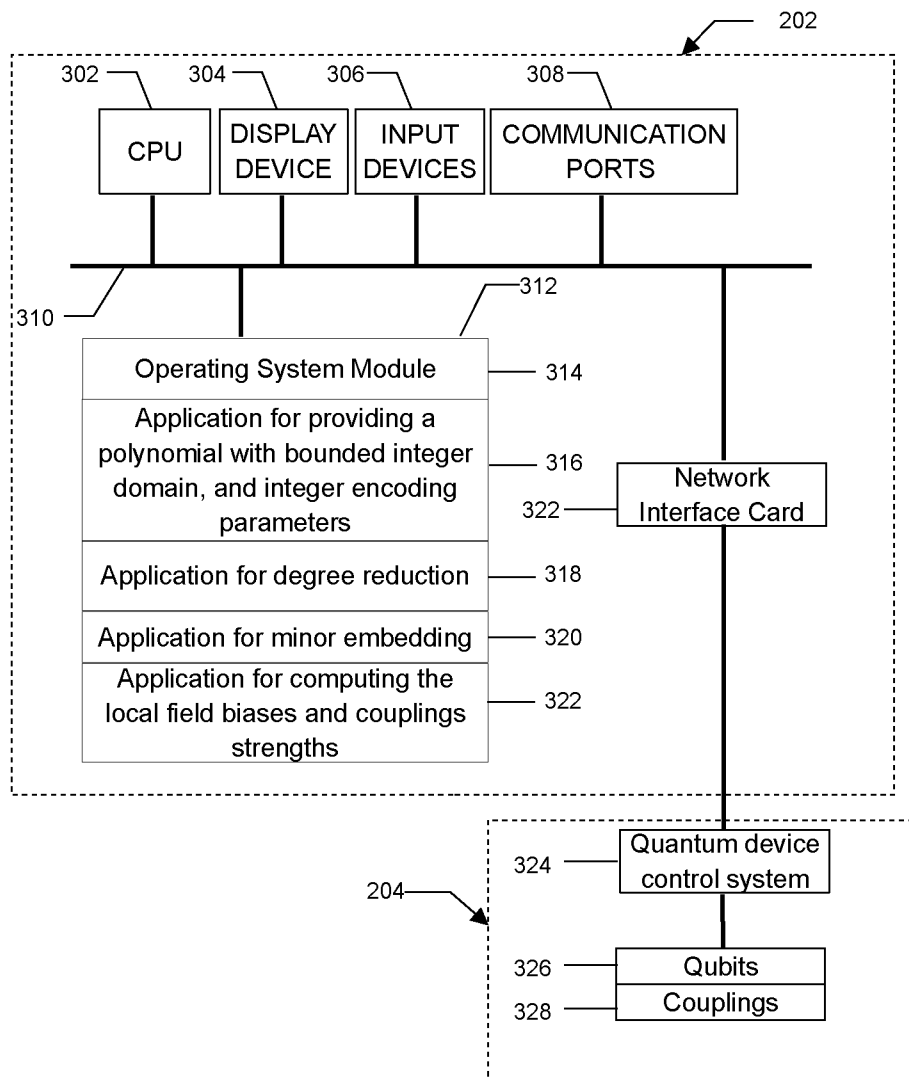


2/7
FIG. 2

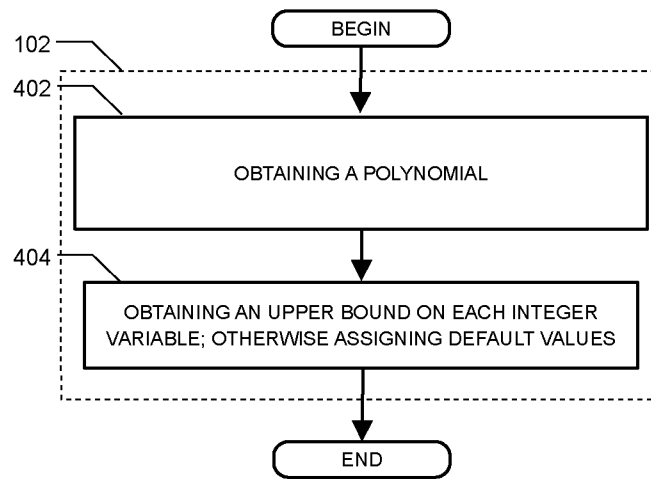


3/7

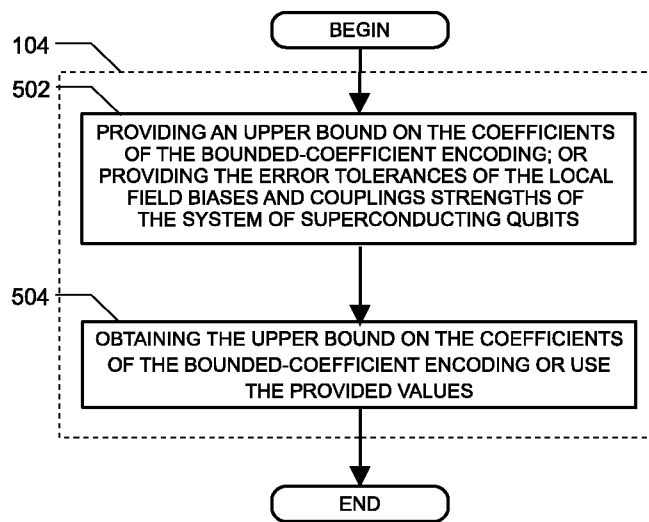
FIG. 3



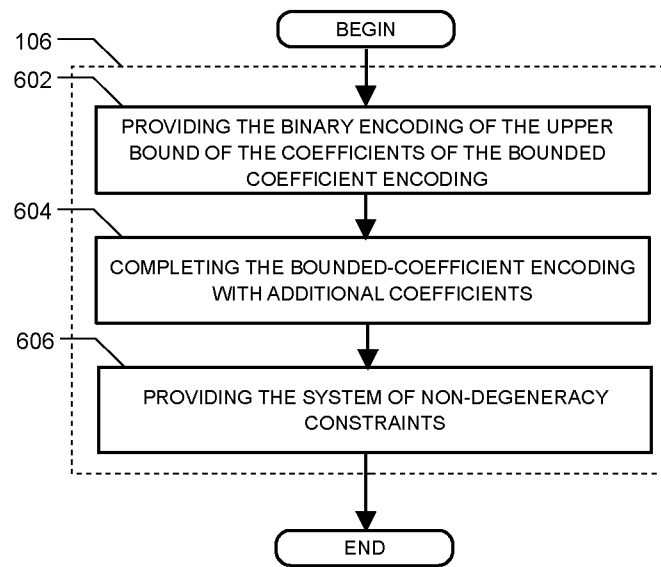
4/7
FIG. 4



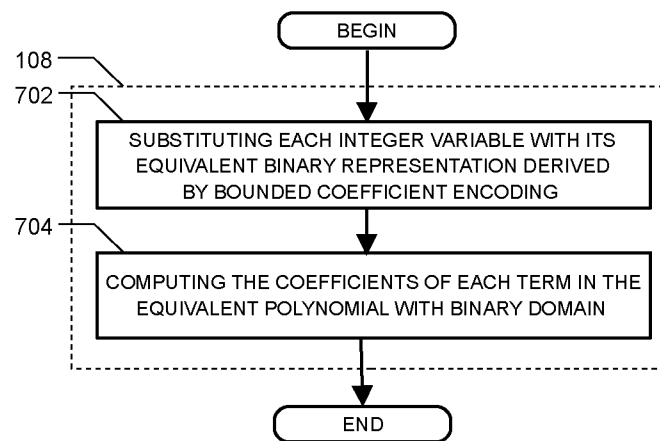
5/7
FIG. 5



6/7
FIG. 6



7/7
FIG. 7



BEGIN

102

OBTAINING A PLURALITY OF INTEGER
VARIABLES ON A BOUNDED INTEGER DOMAIN
AND AN INDICATION FOR A POLYNOMIAL ON
THE OBTAINED INTEGER DOMAIN

104

OBTAINING INTEGER ENCODING PARAMETERS

106

COMPUTING A BOUNDED-COEFFICIENT ENCODING OF
THE INTEGER VARIABLE(S) AND THE SYSTEM OF NON-
DEGENERACY CONSTRAINTS

108

OBTAINING A POLYNOMIAL IN SEVERAL BINARY
VARIABLES EQUIVALENT TO THE PROVIDED
POLYNOMIAL ON A BOUNDED INTEGER DOMAIN

110

PERFORMING A DEGREE REDUCTION ON THE
OBTAINED POLYNOMIAL IN BINARY VARIABLES TO
PROVIDE AN EQUIVALENT POLYNOMIAL OF DEGREE AT
MOST IN SEVERAL BINARY VARIABLES

112

PROVIDING AN ASSIGNMENT OF BINARY VARIABLES OF
THE EQUIVALENT POLYNOMIAL OF DEGREE AT MOST
TWO TO QUBITS

114

SETTING LOCAL FIELD BIASES AND COUPLINGS
STRENGTHS

END