QuantumSCC: A Python Package for Automated Derivation and Diagonalization of Superconducting Circuit Hamiltonians

Rubén Gordillo Hachuel
Department of Physics, Universidad Carlos III de Madrid, Avda. de la Universidad 30, Leganés, 28911 Madrid, Spain
(Dated: January 2025)

Superconducting circuits are a leading platform for quantum computing, offering tunable qubits and fast gate operations. The study of the behavior of these systems is based on the capability of constructing and diagonalizing its quantum Hamiltonian. This task becomes increasingly complex with the emergence of more intricate circuit topologies. To address this need, we introduce QuantumSCC, a Python package that automates the derivation of superconducting circuit Hamiltonians. The algorithm behind this package is based on the approach proposed by Parra-Rodriguez and Egusquiza [1], which employs a geometric framework and the Faddeev–Jackiw symplectic reduction method to systematically construct the Hamiltonian. The program is also capable of diagonalizing the harmonic subspace of the obtained Hamiltonian, finding the natural frequencies of linear circuits. Through detailed examples of linear and nonlinear circuits, we prove QuantumSCC's precision and reliability, showing how its results are consistent with theoretical predictions. Therefore, this package establishes a robust foundation for automating the analysis of superconducting circuits and advancing scalable quantum technologies. Future work will include nonlinear Hamiltonian diagonalization, inclusion of additional circuit elements, and circuit interaction with external electromagnetic fields.

I. INTRODUCTION

Within all existing architectures for the development of qubits, superconducting circuits have emerged as one of the most promising and widely researched hardware platforms for quantum computing [2, 3]. These circuits exploit macroscopic quantum phenomena, such as superconductivity and the Josephson effect, to implement qubits with tunable properties and fast gate operations. The main lines of research in this field include exploring new circuit designs that may improve quantum information encoding [4, 5]; and coupling superconducting qubits with other quantum systems to create hybrid devices with enhanced functionality [6]. Both research topics rely on the ability to construct and diagonalize the quantum Hamiltonian describing the system under study, as this provides insight into the circuit's dynamics and energy spectra.

Traditionally, the derivation of superconducting circuit Hamiltonians was performed manually, using established techniques [7]. However, with the increasing complexity of interesting circuit topologies, this process has become error-prone and time-consuming, particularly for large-scale or highly connected systems. To address these challenges, several open-source software packages have been developed to assist researchers in modeling and analyzing superconducting qubits [8–10]. These tools provide frameworks for simulating superconducting circuits and understanding their dynamics. However, the algorithms behind these packages are based on numerical methods and traditional superconducting circuit theory [7], constructing the Lagrangian from node-flux or loop-charge formulations, and obtaining the Hamiltonian using the Legendre transformation. Although this procedure is effective in some cases, it presents several limitations. First, it may struggle with nonlinear elements such as Josephson junctions, resulting in singularities or cumbersome descriptions that cannot be easily solved [11]. Second, the Hamiltonian formulation in this procedure can encounter singularities, specifically in complex topologies, given the impossibility of carrying out a proper Legendre transformation. Additionally, this method does not differentiate between extended and compact variables, leading to singular cases, particularly when dealing with superconducting islands or loops. Finally, algorithms based on this approach often do not account for the underlying geometric structure, limiting their ability to systematically reduce degrees of freedom or resolve ambiguities in the topological assignments of variables. These weaknesses show the need for a new software package whose algorithm is based on a more robust and algorithmic approach to modeling superconducting circuits.

In this work, we introduce QuantumSCC, a new open-source Python package that automates the derivation of superconducting circuit Hamiltonians. The algorithm developed and implemented in QuantumSCC for Hamiltonian derivation is based on the novel method proposed by Parra-Rodriguez and Egusquiza [1, 12]. This new approach employs a geometric framework and the Faddeev–Jackiw symplectic reduction technique [13] to construct the Hamiltonian systematically from first-order differential equations, allowing quantization to be easily addressed. It also introduces a novel topological assignment for branch variables (extended or compact), which is compatible with classical descriptions. This allows the algorithm to correctly identify the character of each variable, treating them properly, and avoiding inconsistencies. By incorporating this algorithm, QuantumSCC offers an automated and scalable framework for obtaining exact Hamiltonian descriptions of superconducting

circuits, addressing challenges that existing tools cannot and enabling the exploration of increasingly complex systems. Furthermore, the package also incorporates tools for diagonalization of the harmonic part of the resulting Hamiltonian, using the canonical transformation proposed in Ref. [14]. This allows the user to easily study the natural frequencies and energy modes of linear superconducting circuits.

The article is organized as follows. In Section II we discuss the algorithm implemented for the Hamiltonian derivation. Section III contains the discussion of the harmonic Hamiltonian diagonalization. In Sections IV and V we present some examples of linear and nonlinear circuits, respectively, that can be solved using QuantumSCC package. Finally, Section VI contains conclusions and future implementations for this work.

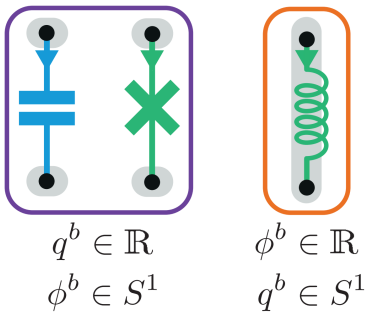

Figure 1. Microscopically-consistent topological ansatz for the branch manifold of lumped elements proposed in Ref. [1] and adopted for the development of the QuantumSCC package. This ansatz assigns branch variables $q^b \in \mathbb{R}$, $\phi^b \in S^1$ to capacitors and Josephson junctions; and the converse for inductors. Image adapted from Ref. [12].

II. HAMILTONIAN DERIVATION ALGORITHM

Deriving the Hamiltonian of a superconducting circuit is essential for understanding its dynamics but can be challenging for circuits with nonlinear elements or complex topologies. QuantumSCC addresses these challenges using an algorithm derived from the novel framework proposed by Parra-Rodriguez and Egusquiza [1, 12].

In this section, we detail the key steps of this algorithm, which forms the backbone of QuantumSCC's analytical capabilities.

A. Geometrical description. Kirchhoff equations

Consider a superconducting circuit composed of B lumped elements, among which there may be capacitors, inductors, and Josephson junctions; arranged in any topology. Each branch of the circuit is defined by a single component, thus having B branches. The electrical state of the circuit can be determined by the flux (ϕ^b) and the charge (q^b) through each branch b of the graph. The system then has $2B$ variables. Circuits containing Josephson junctions may present compact flux variables; hence, the more generic circuit state manifold has the form $\mathcal{M}_{2B} = \mathbb{R}^{2B-k} \times (S^1)^k$, where k is the number of compact variables.

Identifying the compact variables of the system is a critical step of the algorithm, since if the future symplectic form of the Hamiltonian is not written according to all periodicities, the subsequent quantization will result incorrect. However, the spectra of macroscopic quantum flux and charge operators of the different lumped elements have been a topic of intense debate within the QED circuit community for some time now [12]. QuantumSCC implements the microscopically-consistent topological ansatz proposed by Parra-Rodriguez and Egusquiza in Ref. [1]. This ansatz, illustrated in Figure 1, assigns branch variables $q^b \in \mathbb{R}$, $\phi^b \in S^1$ to capacitors and Josephson junctions; whereas inductors are assigned conversely. With this assumption, we will avoid ambiguities in the quantization procedure and ensure that the obtained Hamiltonians faithfully represent the circuit's physical dynamics, as set out in Ref. [1].

To incorporate circuit topology into the formulation, we use Kirchhoff's laws. Kirchhoff's Current Law (KCL) enforces current conservation at each node of the circuit, providing a set of linear equations that relate branch currents. Similarly, Kirchhoff's Voltage Law (KVL) enforces voltage conservation around each independent loop in the circuit, providing constraints on branch voltages. The current and voltage in each branch of the circuit can be expressed as the time derivative of the charge and flux in each branch, respectively. Then, as Kirchhoff's laws apply no matter the actual dynamics of the circuit, we can express them as geometric conditions using charge and flux differential forms:

$$\text{KCL:} \sum_{b\in\mathcal{N}} dq^b = 0, \qquad \text{KVL:} \sum_{b\in\mathcal{P}} d\phi^b = 0, \tag{1}$$

where $b \in \mathcal{N}$ denotes all branches incident on node $\mathcal{N}$ and $b \in \mathcal{P}$ all branches in loop $\mathcal{P}$. This set of equations must be of rank B, according to standard graph theory [15]; and introduces constraints among the $2B$ branch variables, reducing the effective degrees of freedom of the system. An additional property of these equations is that all linear combinations are closed [1]. This means that the Frobenius compatibility conditions [16] are satisfied and the system is integrable. Our goal, therefore, is to integrate these equations, finding a new set of variables that describe the circuit and take into account its geometric structure.

We define the vector of initial flux-charge branch variables as $\boldsymbol{R}^T = (\boldsymbol{\phi}^T, \boldsymbol{q}^T)$; with $\boldsymbol{\phi}^T = (\boldsymbol{\phi}_J^T, \boldsymbol{\phi}_C^T, \boldsymbol{\phi}_L^T)$ and $\boldsymbol{q}^T = (\boldsymbol{q}_J^T, \boldsymbol{q}_C^T, \boldsymbol{q}_L^T)$. Here, the subscripts J, C, L represent the flux-charge variables that come from the Josephson junctions, capacitors, and inductors, respectively. Then, representing the constraints given by Kirchhoff's

laws as a matrix F, we can express the equations (1) as the Pfaff exterior system stated by the expression:

$$\mathsf{F}\, d\boldsymbol{R} = 0, \quad \text{with } \mathsf{F} = \begin{pmatrix} \mathsf{F}_{\text{loop}} & 0 \\ 0 & \mathsf{F}_{\text{cut}} \end{pmatrix}. \tag{2}$$

Here, $d\boldsymbol{R}$ is the differential form of the initial state vector, while F_{loop} and F_{cut} are the matrices that represent, respectively, the KVL and KCL constraints. To construct these matrices, QuantumSCC's algorithm adopts a procedure inspired by the approach presented in Ref. [17]. It begins assembling F_{cut} by inserting a $-1/+1$ in each entry of the matrix that corresponds to the source/destination node of each circuit component, while leaving the rest of the entries equal to zero. The source and destination nodes of each element are chosen by the user, we will see how in section IV. The rank of the resulting F_{cut} matrix is less than its number of rows. Thus, we need to apply a Gauss-Jordan algorithm which, after eliminating the resulting zero rows, transforms the previous matrix into $\mathsf{F}_{\text{cut}} = \begin{pmatrix} \mathbb{I} & \mathsf{A} \end{pmatrix}$. Here, and throughout the paper, $\mathbb{I}$ represents the corresponding identity matrix. Meanwhile, A is a matrix with proper dimensions.

From Ref. [17] we know that F_{loop} and F_{cut} must satisfy the equation: $\mathsf{F}_{\text{loop}}\, \mathsf{F}_{\text{cut}}^T = 0$. Therefore, a valid expression for the matrix F_{loop} is $\mathsf{F}_{\text{loop}} = \begin{pmatrix} -\mathsf{A}^T & \mathbb{I} \end{pmatrix}$, where A is the same matrix that we have found in F_{cut}. It is important to note that the Gauss-Jordan algorithm we applied may need to change the columns of F_{cut} to achieve $\mathsf{F}_{\text{cut}} = \begin{pmatrix} \mathbb{I} & \mathsf{A} \end{pmatrix}$. Then, once we have obtained F_{cut} and F_{loop}, the algorithm simply reorders their columns to recover the initial order and satisfy the expression (2).

Once the matrix F is fully constructed, we can solve the constraints given by equation (2). This means that we need to find a basis for the kernel space of F, such that the matrix K built from these basis vectors satisfies $\mathsf{F}\,\mathsf{K} = 0$. To construct this kernel matrix K, due to the approach we used to build F_{cut} and F_{loop}, we can use the following method [1]:

$$\mathsf{K} = \begin{pmatrix} \mathsf{F}_{\text{cut}}^T & 0 \\ 0 & \mathsf{F}_{\text{loop}}^T \end{pmatrix}. \tag{3}$$

Here, $\mathsf{F}_{\text{cut}}^T$ is the kernel of F_{loop}; and $\mathsf{F}_{\text{loop}}^T$ is the kernel of F_{cut}. However, since the circuit may contain Josephson junctions, we would like to construct the kernel of F_{loop} with the compact flux subspace separated from the rest of the space. To achieve this, following the topological ansatz from Figure 1, we start by expressing the KVL Pfaff system as follows:

$$\mathsf{F}_{\text{loop}}\, d\boldsymbol{\phi} = \mathsf{D}_{\text{loop}} \begin{pmatrix} d\boldsymbol{\phi}_J \\ d\boldsymbol{\phi}_C \end{pmatrix} + \mathsf{G}_{\text{loop}}\, d\boldsymbol{\phi}_L, \tag{4}$$

where D_{loop} is the Kirchhoff matrix of the compact flux subspace; and G_{loop} is the Kirchhoff matrix of the extended flux subspace.

We calculate the kernel of D_{loop}, obtaining a matrix $\mathsf{K}_{\mathsf{D}_{\text{loop}}}$. Then, we add to $\mathsf{K}_{\mathsf{D}_{\text{loop}}}$ a number of zero-rows equal to the number of inductors in the circuit, calling the result $\tilde{\mathsf{K}}_{\mathsf{D}_{\text{loop}}}$. Now, we construct the matrix $\begin{pmatrix} \tilde{\mathsf{K}}_{\mathsf{D}_{\text{loop}}} & \mathsf{F}_{\text{cut}}^T \end{pmatrix}$; and we apply the Gram–Schmidt algorithm [18] to the columns of this matrix. Finally, eliminating the zero-columns that appear, we obtain a valid kernel for F_{loop}, which has the compact flux subspace separated from the rest of the space. We will refer to this kernel matrix as $\tilde{\mathsf{K}}_{\text{loop}}$.

Although this resulting kernel for F_{loop} is valid, there is an additional simplification step our algorithm implements that will have an impact on the Faddeev–Jackiw reduction method (Section II B). Due to the way we have constructed $\tilde{\mathsf{K}}_{\text{loop}}$, there may appear linear-dependent rows in the compact flux subspace. This means that some of the new compact flux variables that describe the circuit have the same dynamics, i.e. a few of them are redundant. In order to make these zero modes explicit in the future symplectic reduction of variables, we perform linear operations between the columns of $\tilde{\mathsf{K}}_{\text{loop}}$, so that the resulting matrix K_{loop} has only one element different from zero in each linearly dependent row. The position of the non-zero element must be the same for each linearly dependent set. With this last step, we get the definitive kernel matrix of F_{loop} that our algorithm uses, K_{loop}. We therefore construct the full kernel matrix K as follows:

$$\mathsf{K} = \begin{pmatrix} \mathsf{K}_{\text{loop}} & 0 \\ 0 & \mathsf{F}_{\text{loop}}^T \end{pmatrix} \quad \text{such that } \mathsf{F}\,\mathsf{K} = 0. \tag{5}$$

We can conclude that the new set of variables, which describe the system taking into account the topological ansatz and the geometric reduction method, are related to the initial set of variables in the following way:

$$\boldsymbol{R} = \mathsf{K}\, \boldsymbol{Z}, \tag{6}$$

where $\boldsymbol{Z}$ is the new flux-charge variable vector defined as $\boldsymbol{Z}^T = \begin{pmatrix} \boldsymbol{\Phi}_c^T & \boldsymbol{\Phi}_e^T & \boldsymbol{Q}^T \end{pmatrix}$. Here, $\boldsymbol{\Phi}_c \in S^1$, are (consistent) local coordinates; while $\boldsymbol{\Phi}_e \in \mathbb{R}$, are global ones. It is important to note that with this entire process we have performed a geometric integration. This means the new variables include unknown constants, which, for simplicity and at the cost of generality, are set to zero. However, they can be adjusted to account for constant external fluxes threading superconducting loops in the circuit.

B. Faddeev–Jackiw symplectic reduction method

The dynamics of the circuit are described by a first-order Lagrangian L. In order to construct this Lagrangian, we start with a pre-Lagrangian function $\tilde{L}$ defined as:

$$\tilde{L} = \frac{1}{2} \sum_{c \in C} q^c \dot{\phi}^c + \frac{1}{2} \sum_{l \in L} \phi^l \dot{q}^l - E(\boldsymbol{\phi}, \boldsymbol{q}), \tag{7}$$

where $\dot{\phi}^c$ and $\dot{q}^l$ represent the time derivative of the flux and charge through the branches $c \in C$ and $l \in L$, respectively; and $E(\boldsymbol{\phi}, \boldsymbol{q})$ is the total energy function of the circuit in terms of the initial flux-charge variables.

The manifold $\mathcal{M}_{2B}$ has an associated degenerate two-form: $\omega_{2B} = \frac{1}{2}(\sum_{j\in J} d\phi^j \wedge dq^j + \sum_{c\in C} dq^c \wedge d\phi^c + \sum_{l\in L} d\phi^l \wedge dq^l)$ [1], which can be rewritten as:

$$\omega_{2B} = \frac{1}{2} d\boldsymbol{R}^T \wedge \Omega_{2B}\, d\boldsymbol{R}, \tag{8}$$

with Ω_{2B} an antisymmetric matrix that contains all the information about the two-form ω_{2B}. Now, as shown in Ref. [1], we can express the function $\tilde{L}$ in terms of this two-form matrix Ω_{2B}:

$$\tilde{L} = \frac{1}{2} \dot{\boldsymbol{R}}^T\, \Omega_{2B}\, \boldsymbol{R} - E(\boldsymbol{\phi}, \boldsymbol{q}), \tag{9}$$

where $\dot{\boldsymbol{R}}$ represents the vector composed of all the time derivatives of the initial flux-charge variables.

The energy function $E(\boldsymbol{\phi}, \boldsymbol{q})$ is constructed as the sum of the energy of each element of the circuit. The energy of a capacitor located in branch c is given by $h_C^c(q^c) = (q^c)^2/2C_c$, with C_c the capacitance of that capacitor; an inductor located in branch l has energy $h_L^l(\phi^l) = (\phi^l)^2/2L_l$, where L_l is the inductance of the inductor; and the energy of a Josephson junction in branch j is $h_J^j(\phi^j) = -E_{J_j} \cos\left(2\pi\phi^j/\Phi_0\right)$, with E_{J_j} the amplitude of the energy function, and Φ_0 the quantum of magnetic flux. Thus, the general expression for the total energy of the circuit can be represented as follows:

$$E(\boldsymbol{\phi}, \boldsymbol{q}) = \sum_{c\in C} h_C^c + \sum_{l\in L} h_L^l - \sum_{j\in J} E_{J_j} \cos\left(\frac{2\pi\phi^j}{\Phi_0}\right) \tag{10}$$

where C, L, J symbolize the set of capacitors, inductors, and Josephson junctions in the circuit, respectively. From this equation (10), we can easily realize that the energy function is made by a quadratic expression (energy of the linear components) plus nonlinear terms (energy of the Josephson junctions). The quadratic part of the function can be formulated in matrix form as follows:

$$\sum_{c\in C} h_C^c + \sum_{l\in L} h_L^l = \frac{1}{2} \boldsymbol{R}^T\, \mathsf{E}_{2B}\, \boldsymbol{R}. \tag{11}$$

Here, E_{2B} is a symmetric matrix that contains the energy information about the inductors and capacitors.

Moreover, we define a set of vectors $\boldsymbol{v}_j$ as an incomplete standard basis, having each vector the same dimension as $\boldsymbol{R}$, but with j going from 1 to the number of Josephson junctions in the circuit. In that way, we can also express the nonlinear energy term as a function of the flux-charge variable vector $\boldsymbol{R}$:

$$\sum_{j\in J} E_{J_j} \cos\left(\frac{2\pi\phi^j}{\Phi_0}\right) = \sum_{j\in J} E_{J_j} \cos\left(\frac{2\pi \boldsymbol{v}_j^T \boldsymbol{R}}{\Phi_0}\right). \tag{12}$$

Now, introducing equations (11) and (12) into the function from expression (9), and applying the reduction variable given by the statement (6), we finally obtain the Lagrangian of the circuit [1], which is expressed in terms of flux-charge variables $\boldsymbol{Z}$:

$$\begin{aligned} L(\boldsymbol{Z}, \dot{\boldsymbol{Z}}) = \frac{1}{2} \dot{\boldsymbol{Z}}^T\, \Omega\, \boldsymbol{Z} - \frac{1}{2} \boldsymbol{Z}^T\, \mathsf{E}\, \boldsymbol{Z} \\ + \sum_{j\in J} E_{J_j} \cos\left(\frac{2\pi \boldsymbol{v}_j^T\, \mathsf{K}\, \boldsymbol{Z}}{\Phi_0}\right), \end{aligned} \tag{13}$$

where $\Omega = \mathsf{K}^T\, \Omega_{2B}\, \mathsf{K}$; and $\mathsf{E} = \mathsf{K}^T\, \mathsf{E}_{2B}\, \mathsf{K}$.

This Lagrangian expression (13) may have zero modes that do not contribute to the circuit dynamics. To eliminate them, we apply the Faddeev–Jackiw symplectic reduction method [13], which also gives us a valid expression for the Hamiltonian of the system, if possible. The method consists of finding a basis change matrix V that transforms the two-form matrix Ω into its canonical form:

$$\mathsf{V}^T\, \Omega\, \mathsf{V} = \begin{pmatrix} \mathsf{J} & 0 \\ 0 & 0 \end{pmatrix} \quad \text{with} \quad \mathsf{J} = \begin{pmatrix} 0 & \mathbb{I} \\ -\mathbb{I} & 0 \end{pmatrix}. \tag{14}$$

Here, J is known as the symplectic matrix.

Since we have built the matrix K_{loop} by simplifying the linear-dependent rows of $\tilde{\mathsf{K}}_{\text{loop}}$ (explained in Section II A), we have already separated the canonical flux variables from those without dynamics in the variable vector $\boldsymbol{Z}$. Hence, we do not need to apply any variable change to the flux subspace, thus avoiding mixing compact and extended fluxes. Now, we have to find the variable change of the charge subspace. To do this, we realize that the two-form matrix Ω, due to the way we have built it, already gives us the change of variable we need:

$$\Omega = \begin{pmatrix} 0 & \Omega_{\phi q} \\ -\Omega_{\phi q}^T & 0 \end{pmatrix} \quad \rightarrow \quad \tilde{V} = \begin{pmatrix} \mathbb{I} & 0 \\ 0 & \Omega_{\phi q} \end{pmatrix}^{-1}, \tag{15}$$

where $\Omega_{\phi q}$ corresponds to the upper right matrix block, which has the same number of rows as flux variables in $\boldsymbol{Z}$, and the same number of columns as charge variables in $\boldsymbol{Z}$. Here, $\tilde{\mathsf{V}}$ is a previous form of the final matrix V that does not yet satisfy the equation (14). To finally construct V, we simply move the rows and columns from $\tilde{\mathsf{V}}$ that correspond to variables without dynamics to the end, getting the matrix V, which does satisfy the expression (14). Therefore, the new variable change can be expressed as

$$\boldsymbol{Z} = \mathsf{V} \begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{w} \end{pmatrix}, \tag{16}$$

with $\boldsymbol{\xi}^T = \left(\boldsymbol{\xi}_\Phi^T \;\; \boldsymbol{\xi}_Q^T\right)$ the vector of the canonical variables, having $\boldsymbol{\xi}_\Phi$ and $\boldsymbol{\xi}_Q$ the same dimension; and $\boldsymbol{w}$ the vector of the variables that corresponds to the zero modes. The vector of flux variables $\boldsymbol{\xi}_\Phi$ can be divided as $\boldsymbol{\xi}_\Phi^T = \left(\boldsymbol{\Phi}_c^T \;\; \boldsymbol{\Phi}_e^T\right)$, where $\boldsymbol{\Phi}_c \in S^1$ represents a vector of (consistent) local flux coordinates; and $\boldsymbol{\Phi}_e \in \mathbb{R}$ represents a vector of global flux variables.

Expressing the Lagrangian in terms of these new variables gives the following total energy function, $\tilde{H}$:

$$\tilde{H}(\boldsymbol{\xi}, \boldsymbol{w}) = \frac{1}{2}\begin{pmatrix}\boldsymbol{\xi}^T & \boldsymbol{w}^T\end{pmatrix}\tilde{\mathsf{H}}\begin{pmatrix}\boldsymbol{\xi}\\ \boldsymbol{w}\end{pmatrix} - \sum_{j\in J} E_{J_j}\cos\left(\frac{2\pi \boldsymbol{v}_j^T\,\mathsf{K}\,\mathsf{V}\left(\boldsymbol{\xi}^T \;\; \boldsymbol{w}^T\right)^T}{\Phi_0}\right), \tag{17}$$

where $\tilde{\mathsf{H}} = \mathsf{V}^T\,\mathsf{E}\,\mathsf{V}$. This expression may depend on variables without dynamics, $\boldsymbol{w}$. Now, according to Ref. [1], if we can express $\boldsymbol{w}$ in terms of $\boldsymbol{\xi}$, the energy function of equation (17) will become a valid expression for the Hamiltonian of the system. To achieve it, the arguments of the cosine functions in $\tilde{H}$ cannot depend on $\boldsymbol{w}$, and we need to solve the following set of differential equations: $\partial\tilde{H}(\boldsymbol{\xi}, \boldsymbol{w})/\partial\boldsymbol{w} = 0$. If any of these conditions cannot be satisfied, the circuit does not present Hamiltonian dynamics, and we cannot know the behavior of the circuit with QuantumSCC's algorithm.

Solving the previous set of differential equations is equivalent to calculating the quadratic Hamiltonian matrix H as:

$$\mathsf{H} = \tilde{\mathsf{H}}_{\xi\xi} - \tilde{\mathsf{H}}_{\xi w}\tilde{\mathsf{H}}_{ww}^{-1}\tilde{\mathsf{H}}_{w\xi}, \quad \text{with} \quad \tilde{\mathsf{H}} = \begin{pmatrix}\tilde{\mathsf{H}}_{\xi\xi} & \tilde{\mathsf{H}}_{\xi w}\\ \tilde{\mathsf{H}}_{w\xi} & \tilde{\mathsf{H}}_{ww}\end{pmatrix}, \tag{18}$$

where $\tilde{\mathsf{H}}_{\xi\xi}$ is the matrix block corresponding to the subspace $\boldsymbol{\xi}^2$, and the rest accordingly. Moreover, $\tilde{\mathsf{H}}_{ww}^{-1}$ represents the Moore–Penrose inverse [19] (or pseudoinverse) of $\tilde{\mathsf{H}}_{ww}$. Finally, with this quadratic Hamiltonian matrix H, we achieve the following expression for the Hamiltonian of the circuit:

$$H(\boldsymbol{\xi}) = \frac{1}{2}\boldsymbol{\xi}^T\,\mathsf{H}\,\boldsymbol{\xi} - \sum_{j\in J} E_{J_j}\cos\left(\frac{2\pi \boldsymbol{v}_j^T\,\mathsf{K}\,\mathsf{V}\begin{pmatrix}\boldsymbol{\xi}\\ \boldsymbol{w}\end{pmatrix}}{\Phi_0}\right). \tag{19}$$

C. Canonical quantization

As the basis change matrix V was constructed using the two-form matrix Ω, the resulting flux variables $\boldsymbol{\xi}_\Phi$ are conjugates of the charge variables $\boldsymbol{\xi}_Q$, in order. This means that they locally satisfy the classical Poisson brackets, which is a globally well-defined entity. Now, to quantize the Hamiltonian, it is precise to differentiate between the two flux topologies. On the one hand, fluxes $\boldsymbol{\Phi}_e \in \mathbb{R}$ and their respective conjugated charges are quantized with the canonical commutation relation: $[\hat{\boldsymbol{\xi}}_\Phi^i, \hat{\boldsymbol{\xi}}_Q^j] = i\hbar\,\mathsf{J}_{ij}$, where J is the symplectic matrix of expression (14). On the other hand, fluxes $\boldsymbol{\Phi}_c \in S^1$ and their respective conjugated charges must be quantized using periodic functions to achieve a globally well-defined expression using local coordinates. Therefore, we employ the number and phase variables, defined as $n = Q/(2e)$ and $\varphi = 2\pi\Phi/\Phi_0$, respectively. Then, we promote the variables to the operators and impose the commutator relation $[\hat{n}, e^{i\hat{\varphi}}] = e^{i\hat{\varphi}}$. With this, we achieve the quantum Hamiltonian of the system:

$$\hat{H}(\hat{\boldsymbol{\xi}}) = \frac{1}{2}\hat{\boldsymbol{\xi}}^T\,\mathsf{H}\,\hat{\boldsymbol{\xi}} - \sum_{j\in J} E_{J_j}\cos\left(\hat{\varphi}_j\right), \tag{20}$$

where $\hat{\varphi}_j = 2\pi \boldsymbol{v}_j^T\,\mathsf{K}\,\mathsf{V}\left(\hat{\boldsymbol{\xi}}^T \;\; \hat{\boldsymbol{w}}^T\right)^T/\Phi_0$ are the phase operators. It is important to note that the quantization of the number operator from the compact flux subspace is not unique. Actually, there are infinite quasiperiodic quantizations. This phenomenon is known as inequivalent quantizations. For more information, see Ref. [1]

III. HARMONIC HAMILTONIAN DIAGONALIZATION

The quantum Hamiltonian of the circuit can be expressed as the sum of three contributions: a harmonic term, which is a quadratic function that depends only on extended fluxes and their respective conjugate charges; a nonlinear term, composed of the sum of a quadratic function that depends only on compact fluxes and their respective conjugate charges plus a nonlinear function; and an interaction term. Therefore, the quantum Hamiltonian function of the expression (20) can be written as:

$$\hat{H}(\hat{\boldsymbol{\xi}}) = \frac{1}{2}\left(\hat{\boldsymbol{\xi}}_e^T\,\mathsf{H}_e\,\hat{\boldsymbol{\xi}}_e + \hat{\boldsymbol{\xi}}^T\,\mathsf{H}_{\text{int}}\,\hat{\boldsymbol{\xi}} + \hat{\boldsymbol{\xi}}_c^T\,\mathsf{H}_c\,\hat{\boldsymbol{\xi}}_c\right) - \sum_{j\in J} E_{J_j}\cos\left(\hat{\varphi}_j\right), \tag{21}$$

where $\hat{\boldsymbol{\xi}}_e$, H_e are the operator vector and the quadratic quantum Hamiltonian matrix corresponding to extended fluxes and their conjugated charges subspace, respectively; H_{int} is the quadratic quantum Hamiltonian matrix corresponding to the interaction space; and $\hat{\boldsymbol{\xi}}_c$, H_c are the operator vector and the quadratic quantum Hamiltonian matrix corresponding to compact fluxes and their conjugated charges subspace, respectively.

Thus, we will now explain how, using the method proposed in Ref. [14], QuantumSCC's algorithm diagonalizes the matrix H_e, finding the natural frequencies of the harmonic subspace and expressing the harmonic Hamiltonian in terms of ladder operators. As this Hamiltonian is quadratic, the equations of motion of the system are

linear and can be written as: $d\hat{\boldsymbol{\xi}}_e/dt = \mathsf{J}\,\mathsf{H}_e\,\hat{\boldsymbol{\xi}}_e$. Here, J is again the symplectic matrix with the proper dimensions. As H_e is a positive semidefinite matrix and we have removed all zero modes, the eigenvalues of $\mathsf{J}\,\mathsf{H}_e$ can only be purely imaginary by conjugate pairs [20]; and their modules are the natural frequencies of the harmonic subspace [14]. Therefore, the symplectic basis change matrix T, which brings H_e to its canonical form N_e, can be constructed with the eigenvectors of the motion matrix $\mathsf{J}\,\mathsf{H}_e$.

We label the eigenvalues as λ_i and $\bar{\lambda}_i$, with eigenvectors g_i^j and $\bar{g}_i^j$, respectively. Here, the index i refers to different eigenvalues and the index j indicates different eigenvectors for the same eigenvalue. Now, in order to construct T, first we need to normalize the eigenvectors under the standard symplectic inner product: $\alpha(g_i^j, \bar{g}_i^j) = (g_i^j)^T\,\mathsf{J}\,\bar{g}_i^j$. To achieve it, we implemented a symplectic Gram-Schmidt algorithm. We start by normalizing the first eigenvector of each eigenvalue ($j = 1$) as follows:

$$e_i^1 = \frac{g_i^1}{\sqrt{\sigma_i \alpha(g_i^1, \bar{g}_i^1)}}, \tag{22}$$

with $\sigma_i = \mathrm{i}\cdot\mathrm{sign}(\alpha(g_i^1, \bar{g}_i^1)/\mathrm{i})$. The rest of the eigenvectors ($j > 1$) are normalized applying the following recursive expressions:

$$\begin{aligned} \tilde{e}_i^j &= g_i^j - \sum_{m=1}^{j-1} \sigma_i \cdot \bar{\alpha}(e_i^m, \bar{g}_i^j) \cdot e_i^m \\ \text{such that} \quad e_i^j &= \frac{\tilde{e}_i^j}{(\alpha(\tilde{e}_i^j, \bar{\tilde{e}}_i^j))^{1/2}}. \end{aligned} \tag{23}$$

Once the normalization process is completed, we can proceed with the construction of the symplectic basis change matrix T as $\mathsf{T} = \begin{pmatrix}\mathsf{T}_+ & \mathsf{T}_-\end{pmatrix}$, where each matrix $\mathsf{T}_\pm$ has structure $\mathsf{T}_\pm = \begin{pmatrix}\mathsf{T}_{1\pm} & \dots & \mathsf{T}_{i\pm} & \dots\end{pmatrix}$; and $\mathsf{T}_{i\pm}$ matrices are built as follows:

$$\begin{aligned} \mathsf{T}_{i+} &= \{\mathbf{t}_i^1, \mathbf{t}_i^2, \dots\} \text{ such that } \mathbf{t}_i^j = \sqrt{2}\cdot Re\{e_i^j\}, \\ \mathsf{T}_{i-} &= \{\mathbf{s}_i^1, \mathbf{s}_i^2, \dots\} \text{ such that } \mathbf{s}_i^j = \sqrt{2}\cdot Im\{e_i^j\}. \end{aligned} \tag{24}$$

Therefore, the basis change that transforms H_e into its canonical form N_e can be expressed as: $\hat{\boldsymbol{\xi}}_e = \mathsf{T}\,\hat{\boldsymbol{\Psi}}$, where $\hat{\boldsymbol{\Psi}}$ is the new operator vector from the extended flux subspace and the respective conjugate charges. However, we would like to express the quantum harmonic Hamiltonian in terms of ladder operators $\hat{a}_i$ and $\hat{a}_i^\dagger$, which satisfy the bosonic commutation relations. Hence, defining the operator vector as $\hat{\boldsymbol{A}}^T = (\hat{a}_1, \dots, \hat{a}_M, \hat{a}_1^\dagger, \dots, \hat{a}_M^\dagger)$, where M is half the dimension of $\hat{\boldsymbol{\Psi}}$, we can find the following basis change matrix:

$$\mathsf{G} = \frac{1}{\sqrt{2}}\begin{pmatrix}\mathbb{I} & \mathbb{I} \\ -\mathrm{i}\mathbb{I} & \mathrm{i}\mathbb{I}\end{pmatrix} \text{ such that } \hat{\boldsymbol{\Psi}} = \mathsf{G}\,\hat{\boldsymbol{A}}, \tag{25}$$

with which we can finally get the diagonalized form of the quantum harmonic Hamiltonian in terms of the ladder operators:

$$\hat{H}_e = \frac{1}{2}\hat{\boldsymbol{A}}^\dagger\,\mathsf{N}_e\,\hat{\boldsymbol{A}} = \sum_{j=1}^{M}\hbar\omega_j\,\hat{a}_j^\dagger\hat{a}_j, \tag{26}$$

where $\mathsf{N}_e = \mathsf{G}^\dagger\,\mathsf{T}^T\,\mathsf{H}_e\,\mathsf{T}\,\mathsf{G} = \mathsf{T}^T\,\mathsf{H}_e\,\mathsf{T}$; ω_j are the natural angular frequencies of the harmonic oscillator; and we have removed the additive constants that represent the ground state energies, since they do not contribute to the dynamics of the system. For simplicity, throughout the paper we will always remove these constants from the quantum Hamiltonian expressions.

IV. LINEAR CIRCUIT EXAMPLES

By linear superconducting circuits, we refer to superconducting circuits composed exclusively of linear elements, i.e. capacitors and inductors. Therefore, the quantum Hamiltonian of such systems does not have compact variables, consisting solely of a quadratic harmonic term: $\hat{H} = \hat{\boldsymbol{\xi}}_e^T\,\mathsf{H}_e\,\hat{\boldsymbol{\xi}}_e$.

In this section, we will explain how the user can employ the QuantumSCC package to construct a linear superconducting circuit, obtaining its diagonalized quantum Hamiltonian in terms of ladder operators, which shows the natural frequencies of the system. Finally, to demonstrate the correct functioning of the package, we will solve a couple of specific examples, comparing the results with those theoretically expected.

The first step is to define the properties of all the components that will be part of the circuit. Therefore, per each different element we need to create an object with the desired characteristics. In the case of an inductor, we need to indicate its value, which can be its inductance or its energy; and the units of the given value, multiples of Henry for inductance values and multiples of Hertz for energy values. For a capacitor, we also have to indicate its value, which can be its capacitance or its energy; and the units, multiples of Farad for capacitance values and multiples of Hertz for energy values. Below, we exemplify the definition of these objects in the code:

```
L = Inductor(value = 1, unit = 'nH')
C = Capacitor(value = 0.1, unit = 'nF')
```

The second step is to define the circuit topology, using the previously defined components. The way to do this is by creating a list of tuples, where each tuple represents a branch of the circuit and has the following structure: [source node, destination node, component]. Then, we use the resulting list to create the object that contains the circuit information. Next, we exemplify the definition of a circuit object for the case of an LC oscillator circuit, using the previously defined components:

```
LC_topology = [(0,1,L), (0,1,C)]
LC_circuit = Circuit(LC_topology)
```

The last step is to print the expression of the diagonalized quantum Hamiltonian that describes the dynamics of the input linear superconducting circuit. For this purpose, we need to apply the function from the class 'Circuit' denoted as 'diagonal_harmonic_Hamiltonian()'. The only parameter of this function is the number of decimal places with which the program prints the natural frequencies of the circuit, which by default is three. The code example for the previously defined LC oscillator circuit is the following:

```
LC_circuit.diagonal_harmonic_Hamiltonian()
```

A. LC oscillator

The first example, as could not be otherwise, is the LC oscillator circuit shown in Figure 2. First, we will show the theoretically expected result and then compare it with the one obtained by the QuantumSCC package. Using variables Φ and Q from Figure 2, the quantum Hamiltonian of this circuit can be expressed as:

$$\hat{H} = \frac{\hat{Q}^2}{2C} + \frac{\hat{\Phi}^2}{2L} \quad \text{with} \quad \left[\hat{\Phi}, \hat{Q}\right] = i\hbar. \tag{27}$$

This circuit has a single natural angular frequency, which can be determined as $\omega = 1/\sqrt{LC}$. Hence, according to expression (26), the expected diagonalized quantum Hamiltonian in terms of ladder operators must be expressed as: $\hat{H} = \hbar\omega\,\hat{a}^\dagger\hat{a}$, where the ladder operators ($\hat{a}$ and $\hat{a}^\dagger$) are related to the flux and charge operators ($\hat{\Phi}$ and $\hat{Q}$) in an equivalent way to that shown in equation (25).

Now, we insert this circuit topology into the QuantumSCC package, choosing an inductance $L = 1$ nH and a capacitance $C = 0.1$ nF. The program then, for a precision of 3 decimal places, returns the following quantum Hamiltonian expression:

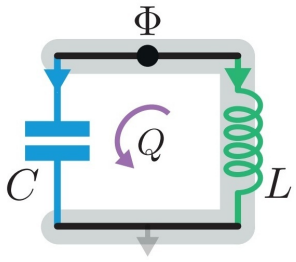

Figure 2. Scheme of an LC oscillator superconducting circuit made up of an inductor of inductance L and a capacitor of capacitance C. The circuit has only one degree of freedom, denoted as variables Φ and Q. Image adapted from Ref. [1].

$$\hat{H} = \hbar\omega_1\,\hat{a}_1^\dagger\hat{a}_1, \tag{28}$$

where $\omega_1 = 3.162$ GHz. Comparing this natural angular frequency returned by the program, but this time with a precision of 15 decimal places, with the theoretically expected result, we obtain a relative error of $\backsim 4.15 \cdot 10^{-7}$. This means that the first deviation between both values is in the sixth decimal place. Therefore, we can conclude that QuantumSCC has returned the correct diagonal quantum Hamiltonian for the LC oscillator circuit.

B. Capacitively coupled LC oscillators

In this example, we capacitively couple two equal LC oscillators from the previous example. The circuit scheme is shown in Figure 3. Here, we designate the inductance and capacitance of the LC oscillators as L and C, respectively; while the capacitance of the coupling capacitor is called C_g.

In order to theoretically obtain the quantum Hamiltonian of this circuit, we follow the node-flux method from Ref. [7], obtaining the following quantum Hamiltonian for the circuit, in terms of variables Φ_1, Q_1, Φ_2 and Q_2 from Figure 3:

$$\begin{aligned}\hat{H} = &\frac{1}{2}\frac{C_\Sigma}{C_\Sigma^2 - C_g^2}(\hat{Q}_1^2 + \hat{Q}_2^2) \\ &+ \frac{C_g}{C_\Sigma^2 - C_g^2}\hat{Q}_1\hat{Q}_2 + \frac{1}{2L}(\hat{\Phi}_1^2 + \hat{\Phi}_2^2),\end{aligned} \tag{29}$$

where $C_\Sigma = C + C_g$. Now, we just have to apply the same diagonalization process employed in QuantumSCC package, and explained in Section III. In that way, the theoretical result for the diagonalized quantum Hamiltonian expression of this circuit is as follows:

$$\hat{H} = \hbar\omega_1\,\hat{a}_1^\dagger\hat{a}_1 + \hbar\omega_2\,\hat{a}_2^\dagger\hat{a}_2, \tag{30}$$

with $\omega_1 = 1/\sqrt{L(C + 2C_g)}$, and $\omega_2 = 1/\sqrt{LC}$. Here, as in the previous example, the ladder operators ($\hat{a}_1$, $\hat{a}_1^\dagger$ and $\hat{a}_2$, $\hat{a}_2^\dagger$) are also related to the flux and charge operators ($\hat{\Phi}_1$, $\hat{Q}_1$ and $\hat{\Phi}_2$, $\hat{Q}_2$) in an equivalent way to that shown in equation (25).

Next, we insert this circuit topology into the QuantumSCC package, choosing an inductance $L = 1$ nH and capacitances $C = 0.1$ nF, $C_g = 0.2$ nF. The program then, for a precision of 3 decimal places, returns the following quantum Hamiltonian expression:

$$\hat{H} = \hbar\omega_1\,\hat{a}_1^\dagger\hat{a}_1 + \hbar\omega_2\,\hat{a}_2^\dagger\hat{a}_2, \tag{31}$$

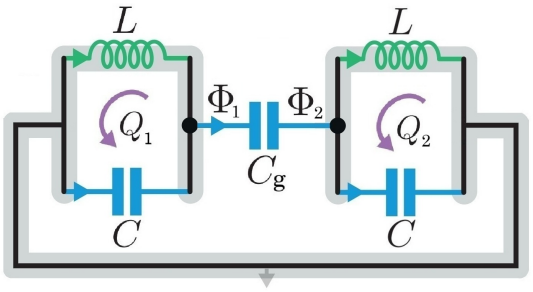

Figure 3. Scheme of a superconducting circuit made up of two LC oscillator coupled capacitively. Both LC oscillators are made up of an inductor of inductance L and a capacitor of capacitance C, while the coupling capacitor has a capacitance C_g. The system has two degrees of freedom, denoted as variables Φ_1, Q_1 and Φ_2, Q_2.

where $\omega_1 = 1.414$ GHz and $\omega_2 = 3.162$ GHz. This result makes physical sense, since we have obtained one frequency that is equal to the corresponding one of an LC oscillator, while the other frequency we have obtained is smaller given that the coupling between oscillators produces a reduction in energy. Now, comparing the natural frequencies returned by the program, but this time with a precision of 15 decimal places, with the theoretically expected results, in both cases we obtain a relative error of $\backsim 4.15 \cdot 10^{-7}$, which is equal to the one obtained in the previous example. This means that the first deviation in each frequency with their corresponding theoretical value is in the sixth decimal place. Therefore, we can finally conclude that QuantumSCC was able to derive the diagonal quantum Hamiltonian expression for this linear superconducting circuit.

V. NONLINEAR CIRCUIT EXAMPLES

In this section, we will complement the previous section explanation about the QuantumSCC circuit design process by adding Josephson junctions, enabling the user to construct nonlinear circuits. Next, we will explain how the QuantumSCC package prints a valid and diagonalizable expression for the quantum Hamiltonian of the corresponding nonlinear circuit. Finally, we will solve a couple of specific examples, comparing the results with those theoretically expected, and demostrating the correct functioning of the package.

In QuantumSCC, to create an object that contains the desired characteristics for a Josephson junction, we must follow a process similar to that of capacitors or inductors, but with a couple of changes. First, since the energy of Josephson junctions is a nonlinear function, by their value we must indicate directly the amplitude of the energy in units of multiples of Hz. Second, in order to treat Josephson junctions, QuantumSCC must associate a capacitor in parallel with them. This is the only limitation of the algorithm, imposed to avoid nonlinear constraint equations during the process. Below, we exemplify the definition of a Josephson junction in the code, along with the definition of its associated capacitor:

```
C_J = Capacitor(value=0.1, unit='nF')
J = Junction(value=1, unit='GHz', cap=C_J)
```

The process of defining the circuit topology is the same as explained in Section IV. The only thing to keep in mind is that for each Josephson junction, the program will add to the circuit the associate capacitor in parallel to the junction. Finally, to print the valid expression of the quantum Hamiltonian, we must apply the function from the class 'Circuit' denoted as 'Hamiltonian_expression()'. The expression printed by this function will express the flux-charge operators in number-phase representation $(\hat{n}, \hat{\varphi})$; and its only input parameter is the number of decimal places with which the program prints the output parameters. It is also important to note that the quantum Hamiltonian printed by this function may not be the only solution; there may be more than one possible solution, as we will see in Section V B.

A. Fluxonium

In this example, we will study the archetypal circuit for building fluxonium qubits [1], shown in Figure 4. The circuit is made up of three branches in parallel, the first contains an inductor of inductance L, the second contains a Josephson junction with amplitude energy E_J, and the last contains a capacitor of capacitance C. The theoretical quantum Hamiltonian expression for this circuit is derived in Ref. [1]:

$$\hat{H} = \frac{\hat{Q}^2}{2C} + \frac{\hat{\Phi}^2}{2L} - E_J \cdot \cos\left(\frac{2\pi\hat{\Phi}}{\Phi_0}\right), \tag{32}$$

where Φ, $Q \in \mathbb{R}$. Now, we insert this circuit topology into the QuantumSCC package, choosing an inductance

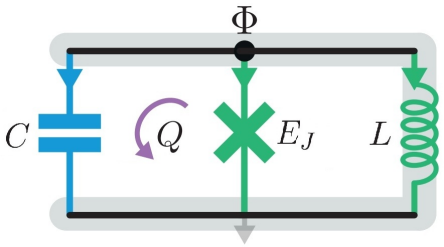

Figure 4. Scheme of the archetypal superconducting circuit for building fluxonium qubits. It is made up of an inductor of inductance L, in parallel with a capacitor of capacitance C, and a Josephson junction with amplitude energy E_J. The circuit has only one degree of freedom, denoted as variables Φ and Q. Image adapted from Ref. [1].

$L = 1$ nH, a capacitance $C = 0.1$ nF, and a Josephson junction energy $E_J = 1$ GHz. The program then, for a precision of 3 decimal places, returns the following quantum Hamiltonian expression:

$$\hat{H}/\hbar = \omega(\hat{\varphi}^2 + \hat{n}^2) - E_J \cdot \cos(b \cdot \hat{\varphi}), \tag{33}$$

with $\hat{n} = \hat{Q}/(2e)$ and $\hat{\varphi} = 2\pi\hat{\Phi}/\Phi_0$, the number-phase operators; $\omega = 3.162$ GHz, the natural angular frequency of the harmonic subspace; $E_J = 1.000$ GHz, the energy of the Josephson junction; and $b = -0.055$. Here, the number-phase operators $\hat{n}$ and $\hat{\varphi}$ belong to the extended flux subspace.

Finally, comparing equations (32) and (33), we can observe that they represent the same Hamiltonian, and therefore they produce the same dynamics. Hence, we can conclude that QuantumSCC was able to find a valid expression for the quantum Hamiltonian of this superconducting circuit.

B. Singular circuit

In this last example, we will study a superconducting circuit which can produce pitfalls if solved using the standard node-flux analysis [12], since this method does not care about the topology of the flux-charge variables. The circuit scheme is shown in Figure 5. It is made up of an inductor of inductance L, in series with a capacitor of capacitance C, and a Josephson junction with amplitude energy E_J. The junction also has in parallel a capacitor of capacitance C_J.

As discussed in Ref. [12], to construct a quantizable Hamiltonian expression that explicitly shows the discrete symmetries, there are only two possible ways to choose the flux-charge variables of the system. Looking at Figure 5, we can realize that these variables are: Φ_1, Q_1 and Φ_2, Q_2; or Φ_a, Q_a and Φ_b, Q_b. Depending on which option is chosen, the theoretical quantum Hamiltonian expression will be slightly different, but both will produce the same dynamics. These theoretical expressions are derived in Ref. [12]:

$$\hat{H} = \frac{(\hat{Q}_1 + \hat{Q}_2)^2}{2C_J} + \frac{\hat{Q}_2^2}{2C} + \frac{\hat{\Phi}_2^2}{2L} - E_J \cdot \cos(\hat{\varphi}_1), \tag{34}$$

$$\hat{H} = \frac{\hat{Q}_b^2}{2C_J} + \frac{(\hat{Q}_a + \hat{Q}_b)^2}{2C} + \frac{\hat{\Phi}_b^2}{2L} - E_J \cdot \cos(\hat{\varphi}_a - \hat{\varphi}_b), \tag{35}$$

where, in both cases, $\hat{\varphi}_i = 2\pi\hat{\Phi}_i/\Phi_0$. Here, the flux operators $\hat{\Phi}_1$ and $\hat{\Phi}_a$ are compact, while $\hat{\Phi}_2$ and $\hat{\Phi}_b$ are extended. As we can notice, no of these Hamiltonians have compact fluxes in the quadratic term. This has allowed us to quantize the compact flux operators using the phase representation, as is explained in Section II C, making explicit the discrete symmetries of the circuit.

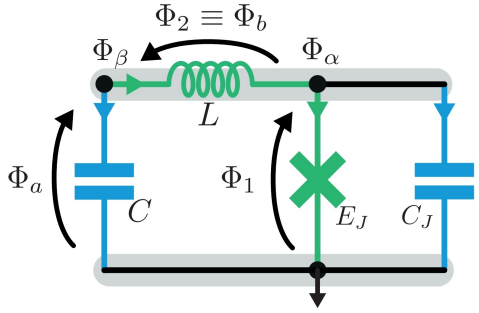

Figure 5. Scheme of a superconducting circuit made up of an inductor of inductance L, in series with a capacitor of capacitance C, and a Josephson junction with amplitude energy E_J. The Josephson junction has in parallel a capacitor of capacitance C_J. The circuit has two degrees of freedom, which can be denoted as variables Φ_1, Q_1 and Φ_2, Q_2; or variables Φ_a, Q_a and Φ_b, Q_b. The branch in which each charge is defined is the same as that of its conjugate flux. Image adapted from Ref. [12].

Now, in order to see if QuantumSCC is capable of returning one of the proper quantized Hamiltonians, we insert this circuit topology into the package, choosing capacitances $C = C_J = 0.1$ nF; inductance $L = 1$ nH; and Josephson junction energy $E_J = 1$ GHz. The program then, for a precision of 3 decimal places, returns the following quantum Hamiltonian expression:

$$\begin{aligned}\hat{H}/\hbar = \omega(\hat{\varphi}_1^2 + \hat{n}_1^2) + b \cdot \hat{n}_1\hat{n}_2 + c \cdot \hat{n}_2^2 \\ - E_J \cdot \cos(d \cdot \hat{\varphi}_2 + f \cdot \hat{\varphi}_1),\end{aligned} \tag{36}$$

with $\hat{n}_i = \hat{Q}_i/(2e)$ and $\hat{\varphi}_i = 2\pi\hat{\Phi}_i/\Phi_0$, the number-phase operators; $\omega = 4.472$ GHz, the natural angular frequency of the harmonic subspace; $E_J = 1.000$ GHz, the energy of the Josephson junction; $b = 0.170$ GHz; $c = 0.016$ GHz; $d = -0.577$; and $f = -0.022$. Here, operators $\hat{n}_1$ and $\hat{\varphi}_1$ belong to the extended flux subspace; while the operators $\hat{n}_2$ and $\hat{\varphi}_2$ belong to the compact flux subspace. Therefore, comparing the expression (36) with the equations (34) and (35), we can realize that the Hamiltonian obtained by QuantumSCC is the same as the one represented in the expression (35). This means that QuantumSCC was able to find one of the two proper quantized Hamiltonians, which explicitly shows the discrete symmetries of the circuit. It is important to note that, as mentioned in Section II C, there are infinite possible quantizations due to the number operator of the compact flux space, $\hat{n}_2$. To represent each of these quantizations, we should substitute in equation (36) this operator with the expression $\hat{n}_2 - \hat{n}_g$, where $\hat{n}_g \in [0, 1)$.

VI. CONCLUSIONS & OUTLOOK

In this work, we have introduced QuantumSCC, a Python package that combines analytical rigor with com-

putational efficiency to automate the derivation and diagonalization of superconducting circuit Hamiltonians. The Hamiltonian derivation algorithm behind this package is based on the geometric framework proposed by Parra-Rodriguez and Egusquiza [1]. It applies the Faddeev–Jackiw symplectic reduction method [13] to identify independent degrees of freedom and construct a consistent Hamiltonian, allowing the quantization process to be easily addressed. This is also due to the topological assignment that QuantumSCC performs for the branch variables of the circuit, differentiating them between extended or compact variables, and allowing the package to address limitations found in existing tools. QuantumSCC is also capable of diagonalizing the harmonic subspace of the resulting Hamiltonians by incorporating the canonical transformation proposed in Ref. [14]. This enables users to easily determine the natural frequencies and energy modes of circuits made up only of linear components.

The validation of QuantumSCC using benchmark examples of linear and nonlinear circuits has demonstrated its precision and reliability, achieving results consistent with theoretical predictions. This makes the package a significant step forward in the automated modeling of superconducting circuits, enabling researchers to tackle increasingly complex circuit designs and paving the way for scalable quantum technologies.

As we mentioned in the introduction, QuantumSCC is an open-source package. In Ref. [21] we linked the GitHub repository where you can find the source code and two notebooks with the examples discussed in this paper and some more.

Further work includes completing the diagonalization algorithm, making QuantumSCC able to diagonalize the Hamiltonian nonlinear subspace and obtaining a full diagonalized expression. This can be performed following the method developed in Ref. [9]. Another future task will be to introduce additional components into the package circuit configuration, such as current sources, voltage sources, and quantum phase slips, which is a trivial process. The last future improvement will be to add the possible interaction between the circuit and an external electromagnetic field. It can be achieved by adding an external and parametrizable magnetic flux per superconducting loop, as is done in Ref. [9].

[1] A. Parra-Rodriguez and I. L. Egusquiza, Geometrical description and faddeev-jackiw quantization of electrical networks, Quantum **8**, 1466 (2024).

[2] J. Clarke and F. K. Wilhelm, Superconducting quantum bits, Nature **453**, 1031 (2008).

[3] J. Q. You and F. Nori, Superconducting circuits and quantum information, Physics Today **58**, 42 (2005).

[4] A. Gyenis, A. Di Paolo, J. Koch, A. Blais, A. A. Houck, and D. I. Schuster, Moving beyond the transmon: Noise-protected superconducting quantum circuits, PRX Quantum **2**, 030101 (2021).

[5] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, A quantum engineer's guide to superconducting qubits, Applied Physics Reviews **6**, 021318 (2019).

[6] Z.-L. Xiang, S. Ashhab, J. Q. You, and F. Nori, Hybrid quantum circuits: Superconducting circuits interacting with other quantum systems, Rev. Mod. Phys. **85**, 623 (2013).

[7] M. H. Devoret, Quantum Fluctuations in Electrical Circuits, in *Fluctuations Quantiques/Quantum Fluctuations*, edited by S. Reynaud, E. Giacobino, and J. Zinn-Justin (1997) p. 351.

[8] P. Groszkowski and J. Koch, Scqubits: a python package for superconducting qubits, Quantum **5**, 583 (2021).

[9] T. Rajabzadeh, Z. Wang, N. Lee, T. Makihara, Y. Guo, and A. H. Safavi-Naeini, Analysis of arbitrary superconducting quantum circuits accompanied by a python package: Sqcircuit, Quantum **7**, 1118 (2023).

[10] P. Aumann, T. Menke, W. D. Oliver, and W. Lechner, Circuitq: an open-source toolbox for superconducting circuits, New Journal of Physics **24**, 093012 (2022).

[11] M. Rymarz and D. P. DiVincenzo, Consistent quantization of nearly singular superconducting circuits, Phys. Rev. X **13**, 021017 (2023).

[12] A. Parra-Rodriguez and I. L. Egusquiza, Exact quantization of nonreciprocal quasi-lumped electrical networks (2024), arXiv:2401.09120 [quant-ph].

[13] L. Faddeev and R. Jackiw, Hamiltonian reduction of unconstrained and constrained systems, Phys. Rev. Lett. **60**, 1692 (1988).

[14] K. Kustura, C. C. Rusconi, and O. Romero-Isart, Quadratic quantum hamiltonians: General canonical transformation to a normal form, Phys. Rev. A **99**, 022130 (2019).

[15] H. Weyl, Repartición de corriente en una red conductora. (introducción al análisis combinatorio), Revista Matemática Hispano-Americana **5**, 153 (1923).

[16] Y. Choquet-Bruhat, *Géométrie différentielle et systèmes extérieurs*, Monographies Universitaires de Mathématiques No. 28 (Dunod, Paris, 1968) Chap. XVII.

[17] G. Burkard, R. H. Koch, and D. P. DiVincenzo, Multilevel quantum description of decoherence in superconducting qubits, Phys. Rev. B **69**, 064503 (2004).

[18] E. Schmidt, Zur theorie der linearen und nichtlinearen integralgleichungen. i. teil: Entwicklung willkürlicher funktionen nach systemen vorgeschriebener, Mathematische Annalen **63**, 433 (1907).

[19] R. Penrose, A generalized inverse for matrices, Mathematical Proceedings of the Cambridge Philosophical Society **51**, 406 (1955).

[20] I. L. Egusquiza and A. Parra-Rodriguez, Algebraic canonical quantization of lumped superconducting networks, Physical Review B **106**, 10.1103/physrevb.106.024510 (2022).

[21] R. Gordillo, QuantumSCC (2025).