



Original software publication

OPENMMF: A library for multimode driven quantum systems

German A. Sinuco-Leon

School of Mathematics and Physical Sciences, University of Sussex, Falmer, BN1 9QH, United Kingdom



ARTICLE INFO

Article history:

Received 30 June 2020

Received in revised form 14 September 2020

Accepted 30 September 2020

Keywords:

Polychromatic driving

Multimode Floquet

Quantum dynamics

Fortran

C++

Python

Dressed states

Non-equilibrium quantum dynamics

ABSTRACT

OPENMMF is a numerical library designed to evaluate the dynamics of quantum systems with a discrete spectrum and driven by an arbitrary combination of harmonic couplings. The time-evolution operator of such systems is calculated as a multifrequency Fourier expansion, which results from expressing the time-dependent Schrödinger equation in the frequency domain (Ho, Chu, and Tietz 1983). The library provides a generic tool to study systems with arbitrary spectral composition, limited only by the available computational resources. OPENMMF includes functionalities to build dense and sparse matrix representations of the Hamiltonian and functions to calculate the micromotion operator and time/phase average of state populations. The library introduces a generalised notion of *dressed state* for systems with polychromatic driving. In this paper, we describe the design and functionality of OPENMMF, provide examples of its use and discuss its range of applicability in problems of current interest in quantum dynamics. The library is written in object-oriented style Fortran90 and includes a set of wrappers for C++ and Python.

© 2020 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Code metadata

Current code version:

v0.4.0

Source code repository:

<https://github.com/OPENMMF>

Legal code license:

CC-BY 4.0

Code versioning system used:

git

Program Title:

OPENMMF

Software code languages, tools, and services used:

Fortran 90, C++, Python

Compilation requirements, operating environments, & dependencies:

LAPACK, MKL-intel, Python 3.7, numpy(1.16.2), ctypes(1.1.0)

Link to developer documentation/manual:

<https://github.com/openMMF/Multimode-Floquet/blob/master/doc/MultimodeFloquet.pdf>
gsinuco@gmail.com

Support email for questions:

1. Motivation and significance

The dynamics of quantum systems driven by polychromatic (or multimode) electromagnetic radiation is of interest to many domains of physical sciences. Over the last few decades, there has been a rapid development of experimental platforms where coherent dynamics of driven quantum systems has been explored with unprecedented precision and sensitivity. Examples of

this include experiments with ultracold atomic ensembles [1,2], trapped ions [3], Nitrogen-Vacancy centres [4–6] and superconducting circuits [7–9]. In all these platforms, the state of an ensemble of quantum systems is manipulated by applying a combination of monochromatic driving fields, produced by lasers or electric/electronic devices. Technical advances of such controlling techniques have led to practical applications (e.g. high-precision spectroscopy [10] or Nuclear Magnetic Resonance [11]), and are instrumental to developing quantum technologies for sensing and computing [12].

E-mail address: gsinuco@gmail.com.

<https://doi.org/10.1016/j.softx.2020.100603>

2352-7110/© 2020 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

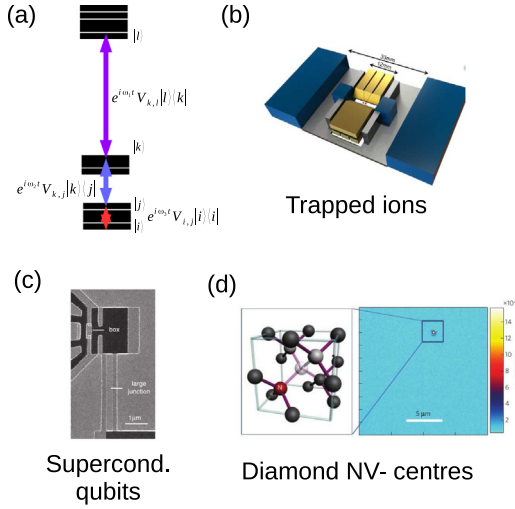


Fig. 1. (a) Schematic energy level structure of a generic quantum system. The basis of states consists of a discrete set of energy states, which define several bands according to the level energy spacing. Inter- and intra-band coupling is induced by electromagnetic radiation tuned at the corresponding frequencies, as indicated by the coupling terms. The wide variety of physical systems described by this model includes (b) trapped ions [3], (c) superconducting qubits [9] and (d) diamond NV-centres [4].

A growing number of applications exploit the possibility to modify the properties of a quantum system through its interaction with electromagnetic radiation. The phenomena of electromagnetic induced transparency (EIT) is a textbook example of this situation [13]: The absorption spectrum of a three-level system is strongly modulated by coupling them through a monochromatic field. Expanding this idea to more complex quantum systems has lead to applications such as robust atomic clocks [14,15], ultra-sensitive field sensors with NV-centres [10], and to discovering new physical phenomena, e.g. topological features in driven many-body systems [16] and time crystals [17], which can be implemented with solid state platforms [18] and ultracold-atomic ensembles loaded into optical lattices [19].

As schematically shown in Fig. 1, the physical systems mentioned above have a similar structure: they are described by a discrete set of states $\{|i\rangle\}$ and polychromatic (or multimode) couplings. The OPENMMF library, introduced in this work, provides tools to evaluate the time-evolution operator of such driven systems. The library implements the multimode Floquet method described originally by Ho, Chu and Tiet in [20], where the semiclassical time-dependent Schrödinger equation is substituted by a time-independent matrix representation in the frequency-domain. The library also provides functions to evaluate physical quantities that cannot be easily accessed from direct integration of the Schrödinger equation in the time-domain, such as the micromotion operator and dressed energies and states [21].

The OPENMMF library includes algorithms that enable its use for a wide variety of problems currently relevant in quantum physics. For example, in the context of quantum technologies, the library can be used to design sequences of electromagnetic pulses for qubit gates [6,22] as well as elements for manipulating matter-waves [23]. The library is also ready to assist emerging areas of research such time-domain quantum simulations [24–26] and (multimode) Floquet engineering [27], where harmonic driving of the system plays an essential role.

2. Time-evolution of multimode driven quantum systems

The main task of the OPENMMF library is to calculate the time-evolution operator, $U(t', t)$, $t' > t$, of systems whose Hamiltonian has the form:

$$H = \sum_{i,j} E_{i,j} |i\rangle \langle j| + \sum_{i,j} \sum_{\ell=1}^D \sum_{n \in \mathbb{Z}} V_{i,j}^{\ell,n} e^{in\omega_{\ell}t} |i\rangle \langle j| + \text{h.c.} \quad (1)$$

where D is the dimension of the Hilbert space, $E_{i,j}$ defines the static component of the Hamiltonian H , $V_{i,j}^{\ell,n}$ is the coupling between the states i and j oscillating at frequency $n\omega_{\ell}$ (i.e. the n th harmonic of the ℓ th fundamental frequency ω_{ℓ}) and N is the number of non-commensurable frequencies.

To calculate the time-evolution operator, we generalise the Rotating (or Resonant) Wave Approximation (RWA) taking into account the complex time dependence of Eq. (1). For this, we rephrase the problem in terms of building a time-dependent unitary transformation, $U_F(t)$ to a new basis $\{|\bar{i}\rangle\}$, that leads to a *time-independent* and diagonal Hamiltonian, \bar{H} . The standard quantum-mechanical transformation rule to the Schrödinger equation [28,29] leads to the condition:

$$U_F^\dagger(t) [H(t) - i\hbar \partial_t] U_F(t) = \sum_{\bar{i}} \bar{E}_{\bar{i}} |\bar{i}\rangle \langle \bar{i}| \quad (2)$$

In the basis of states defined by this transformation, $\{|\bar{i}\rangle\}$, the time evolution operator is diagonal and has the form:

$$\bar{U}(t', t) = \sum_{\bar{i}} e^{-i\bar{E}_{\bar{i}}(t'-t)} |\bar{i}\rangle \langle \bar{i}| \quad (3)$$

which let us to calculate the time evolution operator in the original basis, $\{|i\rangle\}$, just by inverting the transformation $U_F(t)$ according to [29]:

$$U(t', t) = U_F(t') \bar{U}(t', t) U_F(t) \quad (4)$$

To formulate a fully defined computational problem, we express the micromotion operator $U_F(t)$ as the multifrequency Fourier series [20]:

$$U_F(t) = \sum_{\vec{n}} U_{i,i}^{\vec{n}} e^{-i\vec{\omega} \cdot \vec{n}t} |i\rangle \langle i| \quad (5)$$

where $\vec{\omega} = (\omega_1, \omega_2, \dots, \omega_N)$ and \vec{n} is a N -dimensional vector of integers. Using this expansion in Eq. (2) and performing an integral over time, we obtain a fully defined eigenproblem for the eigenvalues $\bar{E}_{\bar{i}}$ and Fourier components of the unitary transformation $U_{i,i}^{\vec{n}}$:

$$\sum_j (E_{i,j} - \hbar \vec{n} \cdot \vec{\omega}) U_{j,i}^{\vec{n}} + \sum_j \sum_{\vec{m}} \left[V_{i,j}^{\ell,n} U_{j,i}^{\vec{n}+\vec{m}} + V_{j,i}^{\ell,n} U_{j,i}^{\vec{n}-\vec{m}} \right] = \bar{E}_{\bar{i}} U_{i,i}^{\vec{n}} \quad (6)$$

where the couplings $V_{i,j}^{\ell,n}$ in Eq. (1) define the values of $V_{i,j}^{\vec{n}}$ with the n -dimensional vector $\vec{m} = (0, \dots, m, \dots, 0)$, where the value m locates at the ℓ -th position. To obtain a finite matrix representation of this problem we truncate the sum over the number of modes of the Fourier expansion Eq. (5), which corresponds to the number of Floquet manifolds for each fundamental frequency. In the Appendix, we show an specific example of the shape of the matrix for a bichromatic driven problem.

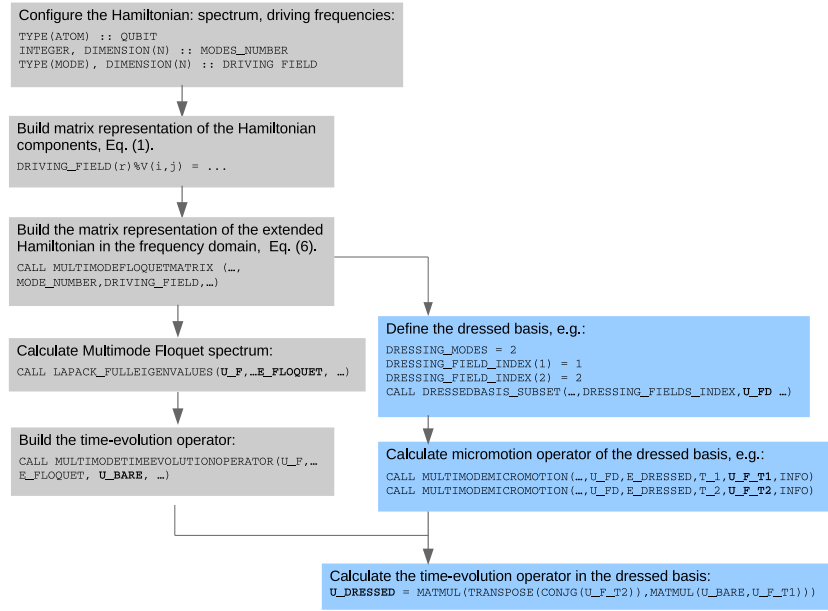


Fig. 2. Typical sequence of function calls to calculate the time-evolution operator of a quantum system with the OPENMMF library. After defining the parameters of the Hamiltonian Eq. (1), the user can build the matrix representation of the Schrödinger equation in the frequency space. The eigenvalues and eigenvectors of this matrix can then be used to evaluate the time-evolution operator. Alternatively, the user can select a subset of the driving fields to define a set of dressed states and evaluate the time-evolution operator in such a basis. Normal and bold fonts indicates input and output subroutine arguments, respectively.

This formulation to calculate the time-evolution operator is equivalent to the multimode Floquet representation of the Hamiltonian that introduces the extended Hilbert space $|E_i, \vec{n}\rangle$ [20, 21,30]. However, the semiclassical description presented here makes emphasis in the experimentally accessible states, which are commonly used to express the static part of the Hamiltonian Eq. (1).

3. Software description

The OPENMMF library is written in object-oriented style FORTRAN 90 and includes wrappers to enable its use with C++ and Python. The software architecture resembles a typical sequence of operations enabled by this library, which is shown in Fig. 2.

The starting point to use the OPENMMF library using any of these languages consists on defining the parameters needed to build a matrix representation of the Hamiltonian components of Eq. (1). Then, the user can execute a series of tasks that lead to the time-evolution operator.

In Fortran, the components of the library can be classified into four groups:

1. Fortran modules that define as global variables relevant physical constants (`physical_constants`) and properties of atomic systems (`ATOMIC_PROPERTIES`).
2. Two Fortran derived data types (and corresponding C++ structures) that define the parameters required to build a matrix representation of the system's Hamiltonian (`TYPE::ATOM` and `TYPE::MODE`).
3. A set of computational routines that perform distinct computational tasks, as in Fig. 2.
4. A set of driver routines, each one to call a sequence of computational routines for solving standard problems with a single subroutine call.

To give a brief overview of the library structure and functionalities, we describe the last three groups. We refer the reader to the user manual for detailed information about the Fortran modules defined in the library, usability of the C++ and Python wrappers and performance of the library [31].

3.1. Setting the system configuration

The parameters required to build a matrix representation of the Hamiltonian Eq. (1) must be initialised using the following three data structures:

1. The Fortran derived type `TYPE::ATOM`, which contains one component to set the dimension of the Hilbert space (`D_Bare`) and another one (declared as an array of type `double`), which specifies the energy spectrum of the system.
2. An array of `integer` type, with as many components as the number of fundamental driving frequencies plus one (i.e. $N + 1$). The entries of this array indicate the number of harmonics of each driving frequency, with the first component corresponding to the static component in Eq. (1).
3. An array of the Fortran derived type `TYPE::MODE`, with as many elements as the total number of driving frequencies plus one (i.e. $1 + \sum_{\ell=1}^N n_{\ell}$). The components of this derived data type include the driving angular frequency (ω), complex matrices to store the explicit representation of the coupling matrix (V), and the number of modes of the Fourier expansion `N_Floquet`.

3.2. Typical computing sequences

Using the previous three data structures and an explicit representation of the components of the Hamiltonian, the user can pro-

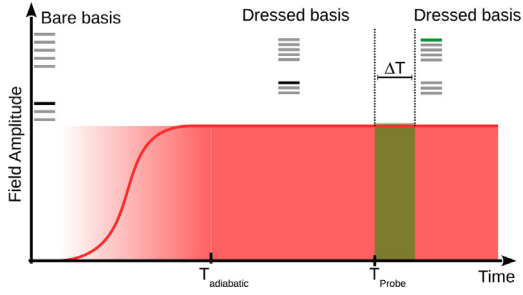


Fig. 3. Typical time sequence to prepare and probe dressed states. The dressing modes are adiabatically switched on and kept constant during the probing period.

ceed to build a finite matrix representation of the eigenproblem Eq. (6). This task can be done using the subroutine MULTIMODEFLOQUETMATRIX, which creates a square-matrix representation of the right-hand-side of Eq. (5).

The eigenvalues and eigenstates of the extended representation of the Hamiltonian can then be obtained using standard diagonalisation routines. The library includes wrappers to call the relevant subroutines of the LAPACK and the MKL-Intel libraries. However, the user is free to use other tools. The resulting eigenvalues and eigenvectors are then employed to calculate the time-evolution ($U(t', t)$, Eq. (3)) and micromotion ($U_F(t)$, Eq. (5)) operators, using the routines MULTIMODETIMEEVOLUTIONOPERATOR and MICROMOTIONOPERATOR, respectively.

The OPENMMF library includes the driver subroutine TIMEEVOLUTIONOPERATOR, which allows the user to perform these three tasks (building the Hamiltonian representation, diagonalisation and evaluation of $U(t', t)$) with a single subroutine call after defining the components of the Hamiltonian.

To take advantage of the sparse nature of Eq. (6), the library also allows us to create a sparse representation of this equation and diagonalisation with the MKL-intel library. The names of the corresponding subroutines have appended the letters _SP, and the user must ensure the correct allocation of memory.

3.3. Time-evolution in a dressed basis

The OPENMMF library provides tools to evaluate the dynamics of driven quantum systems in the so-called *dressed basis*. Fig. 3 presents a situation where it is convenient to express the time evolution operator on a time-dependent basis that differs from the one used to represent the Hamiltonian.

In this case, the driving modes applied to the quantum system can be classified into two groups: dressing and probing modes. Initially, the dressing modes are adiabatically switched on over a period of time $T_{\text{adiabatic}}$, after which their amplitudes and frequencies remain constant. This procedure is shown schematically in Fig. 3, with the red line and the colour gradient indicating the slow change of the amplitudes of the dressing modes. The adiabatic change of the dressing modes ensures that the basis of the static system (on the left-hand-side of Fig. 3) deforms into

the time-dependent basis defined by the transformation $U_F(t)$, Eq. (6) defined by *only* the dressing modes. Thus, for example, when at $T = 0$ the system is prepared in the state with the i th energy, $|i\rangle$, after $T_{\text{adiabatic}}$, this state becomes the corresponding linear superposition associated with the i th eigenvalue \tilde{E}_i .

After a period of evolution driven by the dressing modes, the system is probed by pulses of the probing subset, while the dressing modes are kept constant. During the duration of the probing pulses, $t \in [T_{\text{Probe}}, T_{\text{Probe}} + \Delta T]$, the time-evolution of the system is determined by a Hamiltonian of the form Eq. (1), which must include *both* the dressing and probing modes. The OPENMMF evaluates the time evolution operator initially in the bare basis and includes functions to define the micromotion operator associated with the subset of dressing fields and the corresponding transformation of the time-evolution operator. An example of this situation is shown in the following section.

3.4. Functionality for spin systems

Spin systems are of particular interest due to the simplicity of their energy spectrum and their experimental feasibility with a diverse set of experimental platforms. For these systems, the couplings to harmonic drivings are of the form:

$$H^{\ell,n} = \sum_{i \in x,y,s} e^{n\omega_\ell t + \phi_i^{\ell,n}} V_i^{\ell,n} S_i \quad (7)$$

where S_i is the i th Cartesian component of angular momentum operator, whose matrix representation follows the standard definition in Sec. 3.5 of [32]. Each one of these terms is defined by six real parameters: three phases ($\phi_i^{\ell,n}$) and three amplitudes ($V_i^{\ell,n}$), which are included as components of the Fortran data structure TYPE: :MODE. The OPENMMF library includes the subroutine SETHAMILTONIANCOMPONENTS that uses these parameters to define Hamiltonian components Eq. (7).

The library can also handle the energy configuration of systems composed by two spins S_1 and S_2 , coupled via the dipole interaction $H_{\text{dipole}} = A \mathbf{S}_1 \cdot \mathbf{S}_2$ and each spin interactions as above in Eq. (7). The library evaluates the Hamiltonian and time-evolution operator in the eigenbasis of the dipole interaction, with the energy scale defined by the dipole coupling constant A . The library defines the matrix representation of the coupled system using the uncoupled basis $|S_1, m_1\rangle |S_2, m_2\rangle$ and the definitions in Sec. 3.5 of [32]. For details of the implementation, we refer the reader to the User Guide [31].

This type of system of coupled system is present in the electronic ground states of alkali atoms [33] and single-charged atomic ions [34], which are currently the focus of intense experimental research in quantum technology. The interaction between the nuclear and electronic magnetic dipole moments defines two sub-sets of states separated by an energy gap of order A , which has enabled the use of radio-frequency and microwaves for coherent control of quantum states [33,34]. The library let us to readily evaluate the time-evolution of the atomic states in such situations [35].

4. Illustrative examples

The main features of the library are illustrated with two examples, which are relevant across many experimental platforms. We present a selection of examples that illustrate the potential of OPENMMF for a wide range of applications. The example in Section 4.1 focuses on a two-level systems, which can be extended to study quantum control of quantum systems with a small number of constituents. In Section 4.2 we study a driven tight-binding Hamiltonian, which is of current interest in the context of ultracold atomic physics and Floquet engineering [36].

4.1. Rabi oscillations of a dressed qubit

As our first example, we consider the time-evolution of a two-level system (or qubit) driven by two harmonic drives with the Hamiltonian:

$$H = \hbar\omega_0 S_z + \hbar\Omega_X \cos(\omega_D t + \phi_D) S_x + \hbar\Omega_{P,X} \cos(\omega_P t + \phi_{P,X}) S_x + \hbar\Omega_{P,Z} \cos(\omega_P t + \phi_{P,Z}) S_z \quad (8)$$

where the subindice D (P) indicates that the mode is a Dressing (Probing) the qubit, in the sense defined in Section 3.3.

In this case, the data structures required to setup the matrix representation of the Hamiltonian components are declared with the instructions:

```
TYPE(ATOM)          QUBIT
INTEGER, DIMENSION(3) :: MODES_NUMBER
TYPE(MODE), DIMENSION(3) :: DRIVING_FIELD

MODES_NUMBER = (1,1,1)
```

The second and third components of the integer array `MODES_NUMBER` indicate that the qubit is driven by one harmonic of each frequency. The parameters of the drivings are defined via the components of the `MODE` data type `DRIVING_FIELD`. Since a qubit is a spin 1/2 system, an explicit declaration of the phase and amplitude of the driving fields is sufficient to define the Hamiltonian components. For driving mode D this is done as follows:

```
DRIVING_FIELD(2)%X      = OMEGA_X
DRIVING_FIELD(2)%phi_x  = 0.0
DRIVING_FIELD(2)%omega  = OMEGA_D
```

with a similar set of instruction to define the parameters of the P driving mode.

The calculation of the time-evolution operator in the bare basis is then done following the sequence of instructions detailed in Section 3.2:

```
!=====
!== MULTIMODE TIME-EVOLUTION OPERATOR IN THE BARE BASIS
!=====
CALL SETHAMILTONIANCOMPONENTS(...,MODES_NUM,DRIVING_FIELD,INFO)
CALL MULTIMODEFLOQUETMATRIX(...,MODES_NUM,DRIVING_FIELD,INFO)
CALL LAPACK_FULLEIGENVALUES(U_F,SIZE(U_F,1),E_FLOQUET,INFO)
!=====
!===== EVALUATE TIME-EVOLUTION OPERATOR BETWEEN T1 AND T2 IN THE BARE BASIS
!=====
CALL MULTIMODETIMEEVOLUTINOPERATOR(...,MODES_NUM,U_F,E_FLOQUET,...,DRIVING_FIELD, &
& T1,T2,U_BARE,INFO)
```

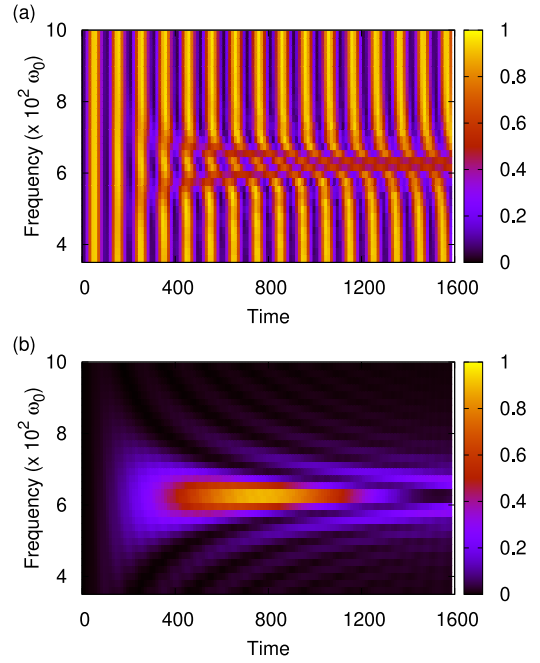


Fig. 4. Probability of transition of a resonantly dressed qubit as a function of the time and probe frequency ω_P (a) in the basis of bare states (b) in the dressed basis defined. The Hamiltonian parameters are: $\hbar\omega_0 = 1.0$, $\hbar\omega_D = 0.025$, $\hbar\omega_P = 1.0$, $\hbar\Omega_{P,X} = 0.005$, $\hbar\Omega_{P,Z} = 0.025$, with all phase factors set to zero.

where the dots represent arguments omitted for shortness and clarity and the bold font indicates output subroutine arguments. This procedure results in the 2×2 matrix `U_BARE` containing the time evolution from T_1 to T_2 .

In this type of systems, it is of interest to calculate the time-evolution operator in the basis of dressed states defined by one of the modes, here selected as the 2nd mode (D). The openMMF library let us define the Fourier decomposition of the required change of basis just by defining the subset of dressing modes as follows:

```
DRESSING_MODES = 2 ! including the static mode
DRESSING_FIELD_INDEX(1) = 1 ! The static mode
DRESSING_FIELD_INDEX(2) = 2 ! The dressing mode
CALL DRESSED BASIS_SUBSET(...,DRESSING_FIELDS_INDEX,
MODES_NUM,DRIVING_FIELD,U_FD,E_DRESSED,...)
```

As a result of this sequence of instructions, we obtain a matrix with the Fourier decomposition of the micromotion operator associated with the subset of dressing modes, `U_FD`, along with an array containing the corresponding dressed eigenenergies `E_DRESSED`. To calculate the time evolution in the dressed basis, `U_DRESSED`, we should evaluate the micromotion operator at times T_1 , $U_F T_1$, and T_2 , $U_F T_2$, which are then used in the transformation Eq. (4). This is done with the following sequence of instructions:

```
CALL MULTIMODEMICROMOTION(...,U_FD,E_DRESSED,T_1,U_F_T1,INFO)
CALL MULTIMODEMICROMOTION(...,U_FD,E_DRESSED,T_2,U_F_T2,INFO)
U_DRESSED = MATMUL(TRANPOSE(CONJG(U_F_T2)),MATMUL(U_BARE,U_F_T1))
```


where the first and second lines produced the time-dependent micromotion operators of the dressed basis, and the third line executes the basis transformation.

As a concrete example of the convenience of the dressed basis, in Fig. 4 we plot the time evolution of the probability of remaining in an initially pure state as a function of time and the frequency of the second driving field. Fast oscillations of this probability are displayed in the original qubit basis, while a slower dynamics is observed in the dressed basis. Note the resonant feature at the frequency $\omega_p = \Omega_x/2.0$ [21].

4.2. Harmonically driven 1-D tight-binding model

Technical advances in atomic and optical physics have enabled experiments with ultracold atomic ensembles loaded in spatially periodic potentials landscapes defined by optical lattices [27]. Controlled temporal modulation of the lattice parameters can induce non-equilibrium phenomena of interest for quantum simulations, and the experimental demonstration of such phenomena is leading to rapid advances in this area. Many important dynamical features of such driven systems are diagnosed via the dependence of eigenvalues and eigenvectors of the Hamiltonian in the frequency-domain with respect to the variation of the driving parameters [37,38].

To illustrate the use of the OPENMMF library for these problems, here we consider a spatially periodic one-dimensional lattice with a time-periodic modulation of the next-neighbour hopping. The lattice has $D/2$ unit cells, each containing two sites with energy offsets $\pm V_0$. The corresponding Hamiltonian reads:

$$H = \sum_{m=1}^D (-1)^m V_0 |m\rangle \langle m| + \sum_{m=1}^{D-1} t e^{i \cos(2\pi m/D)} |m\rangle \langle m+1| + \text{h.c.} + \sum_{m=1}^{D-1} V \cos(\omega t + \phi) |m\rangle \langle m+1| + \text{h.c.} \quad (9)$$

where the first terms defines the energy offset at each lattice site, the second line defines a site-dependent tunnelling rate between neighbouring sites and the last term defines a uniform (i.e. site-independent) modulation of the near-neighbour tunnelling.

The three Fortran data structures containing the parameters of the Hamiltonian are initialised as before, but here only two modes are needed (the static component plus one driving mode). The static and driving components of the Hamiltonian are declared with the following set of instructions:

```
DO m=1,D ! ALL SITES
  DRIVING_FIELD(1)%V(m,m) = ((-1)**m)*V_0
END DO

! NEAR NEIGHBOURS COUPLINGS
DO m=1,D-1
  ! STATIC COMPONENT
  DRIVING_FIELD(1)%V(m,m+1) = t*EXP(DCMPLX(0.0, 1.0)*COS(pi*2*m/D))
  DRIVING_FIELD(1)%V(m+1,m) = t*EXP(DCMPLX(0.0, -1.0)*COS(pi*2*m/D))

  ! DRIVING MODE
  DRIVING_FIELD(2)%V(m,m+1) = V
  DRIVING_FIELD(2)%V(m+1,m) = V
END DO
DRIVING_FIELD(2)%omega = omega
```

which displays a correspondence with the Hamiltonian Eq. (9). Just two more instructions are sufficient to evaluate the eigenvalues and eigenenergies of the frequency-domain representation of the Hamiltonian:

```
CALL MULTIMODEFLOQUETMATRIX(..., MODES_NUMBER, DRIVING_FIELD, INFO)
CALL LAPACK_FULLEIGENVALUES(U_F, ..., E_FLOQUET, INFO)
```

The second line calls a wrapper to a diagonalisation subroutine from the LAPACK library. The resulting eigenvectors are stored in the matrix U_F and the eigenvalues in the array $E_FLOQUET$. A typical example of this calculation is shown in Fig. 5, where the set of eigenenergies are plotted as a function of the driving frequency.

In addition to evaluation of the Floquet spectrum, the numerical calculation of the time-evolution operator is of interest for Hamiltonian engineering with atomic ensembles in atomic lattices [36]. This capability of the OPENMMF can be used to evaluate corrections to analytical approximations of the Effective Hamiltonian [36], and their effects on the dynamics of the atomic cloud [39].

5. Impact

Considering that the Hamiltonian in Eq. (1) describes a wide diversity of physical systems, the OPENMMF library constitutes a general-purpose tool for studying quantum dynamics of driven systems. The library is open-source and provides a ready-to-use interface, which can be exploited by users with a broad range of expertise ranging from graduate students to established researchers. An important feature of the library is the set of functions that allow the user to evaluate the time-evolution on a dressed basis, which is of current interest across different fields. To illustrate the degree of generality of the library, and therefore its potential impact, here we mention three specific research areas where the OPENMMF library can find the direct application.

Coherent control of quantum systems

Precise control over the state of a quantum system is key for quantum technologies. An important operational step in such applications is the precise creation of quantum superposition, which is generally achieved by interleaving periods of free and driven evolution described by a Hamiltonian of the form Eq. (1) [40]. Decoherence induced by the interaction of the system with the environment is a major obstacle to achieve the levels of control required for the realisation of practical applications. However, superposition of dressed states can be more robust to environmental noise than bare states [41,42]. This mechanism to reduce the sensitivity of a quantum system to noise is known as Continuous Dynamical Decoupling and, to be implemented, it requires an experimental sequence that follows the path shown in Fig. 3. This technique has found application improving the

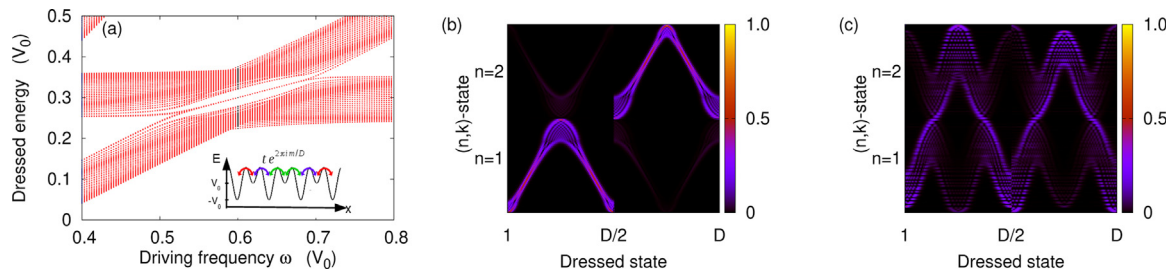


Fig. 5. Dressed spectrum of a driven lattice model. (a) Eigenenergies of the frequency-domain representation of the Hamiltonian Eq. (9) as a function of the driving frequency ω . (inset) Periodic potential described by the tight-binding model Eq. (9). Projection of the dressed states onto the eigen-basis of the static lattice when the driving frequency is (b) red detuned ($\omega = 0.4$, black dots in panel (a)) (c) resonant with the band gap ($\omega = 0.6$, blue dots in panel (a)). The parameters of the model are: $D = 256$, $V_0 = 0.25$, $t = 0.125$, $V = 0.0625$ and $\phi = 0$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

coherence of atomic clocks [14,43] and qubit gates [34,44], as well as for the sensitivity and spatial resolution of low-frequency magnetometers operating [8,10,45].

Time-domain quantum simulations

The frequency-domain representation of the Hamiltonian Eq. (1) with N mutually irrational frequencies is given by Eq. (6). This representation can be interpreted as the Hamiltonian of an N -dimensional lattice with sites label by the index \vec{n} , a DC force equal to $\vec{\omega}$ and couplings between sites controlled by the driving modes. The typical signatures of dynamics of the simulated lattice (e.g. band structure, conductivity, edge-states, etc.) arise in the time-evolution of the system, which is directly accessible with the OPENMMF library. Topologically non-trivial states [25,46] and Anderson localisation in quasi-crystals [26] are two phenomena accessible with this scheme for quantum simulations, which could be realised with a significantly simplified experimental setup.

Out-of-equilibrium phases of driven interacting systems

The behaviour of driven systems can be qualitatively very different from their static counterparts. Various driving schemes have been recently used to engineer effective Hamiltonian displaying out-of-equilibrium phases that are impossible to realise with static systems [47]. Such applications are subject of intense experimental and theoretical research with ultracold atomic gases loaded in optical lattices [36], where the Hamiltonian and driving schemes are similar to those in the example in Section 4.2. In the single-particle case, the dynamics in the limits of high frequency or strong and resonant driving are relatively well understood [39], and novel phases of matter in such regimes have been proposed and realised in recent years [1,37]. However, there are several fundamental questions still to be addressed in the context of many-body driven systems, which emerge from the interplay between interactions, driving and symmetries/disorder [26,48]. The OPENMMF library provides a tool to investigate such phenomena, as we show in our last illustrative example. Though current computational limitations allows only to investigate systems with relatively small Hilbert space dimension, such studies provide a

solid understanding of the non-equilibrium dynamics of driven systems [49–51] and the library provides a platform of simple usability.

6. Conclusions

In this paper, we introduce the OPENMMF library as a tool to calculate the time-evolution operator of quantum systems with discrete spectrum driven by harmonic couplings. The library implements the multimode Floquet expansion of a time-dependent Hamiltonian and provides several functionalities relevant for its analysis. In particular, the library allows the user to evaluate the micromotion operator associated with arbitrary drivings built as a superposition of harmonic terms. In passing, this procedure generalises the notion of dressed states to such situations.

The general formulation implemented in the library enables its use for applications where determining the time-evolution operator is relevant, such as designing optimal control sequences, tailoring the response of quantum systems and studying non-equilibrium quantum dynamics and Hamiltonian engineering. The code is written in Fortran 90, and we provide wrappers for C++ and Python, making it widely accessible and easy usage.

Data statement

All relevant data presented in this work can be generated with the examples of the OPENMMF library (examples/FORTRAN, examples/CPP and examples/PYTHON), which are available via the Elsevier/SoftwareX and the OPENMMF git repositories. The library source code is available as an open-source project through the same channels, covered by a CC-BY 4.0 license.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We acknowledge fruitful comments and input from Pedro Nevado Serrano, Juan Totero Gongora and Barry Garraway. This work has been supported by the EPSRC, United Kingdom grants EP/I010394/1, EP/M013294/1 and the University of Sussex, United Kingdom. Special acknowledgements to Camilo Guerrero and ConectLabs SAS (<https://www.facebook.com/conectalabs/>) for supporting code review of this project.

Appendix. Typical example of the matrix representation of the multimode Hamiltonian

A typical shape of the matrix representation of the right-hand-side of Eq. (6). Here we consider a static system H_0 driven by the three frequencies: $\omega_1, 2\omega_1, \omega_2$. The integer array describing this set of frequencies must be (1, 2, 1). For the two fundamental frequencies, the number of Fourier modes in the decomposition of the micromotion operator is chosen to be $N_F = 3$. With this, the driving of ω_1 and its first harmonic leads to the matrix:

$$\mathcal{H}_1 = \begin{pmatrix} H_0 + 3\hbar\omega_1 & V^{1,1} & V^{1,2} & 0 & 0 & 0 & 0 \\ V^{1,1\dagger} & H_0 + 2\hbar\omega_1 & V^{1,1} & V^{1,2} & 0 & 0 & 0 \\ V^{1,2\dagger} & V^{1,1\dagger} & H_0 + \hbar\omega_1 & V^{1,1} & V^{1,2} & 0 & 0 \\ 0 & V^{1,2\dagger} & V^{1,1\dagger} & H_0 & V^{1,1} & V^{1,2} & 0 \\ 0 & 0 & V^{1,2\dagger} & V^{1,1\dagger} & H_0 - \hbar\omega_1 & V^{1,1} & V^{1,2} \\ 0 & 0 & 0 & V^{1,2\dagger} & V^{1,1\dagger} & H_0 - 2\hbar\omega_1 & V^{1,1} \\ 0 & 0 & 0 & 0 & V^{1,2\dagger} & V^{1,1\dagger} & H_0 - 3\hbar\omega_1 \end{pmatrix}$$

and the full matrix results in:

$$\mathcal{H} = \begin{pmatrix} \mathcal{H}_1 + 3\hbar\omega_2 & \mathcal{V}^{2,1} & 0 & 0 & 0 & 0 & 0 \\ \mathcal{V}^{2,1\dagger} & \mathcal{H}_1 + 2\hbar\omega_2 & \mathcal{V}^{2,1} & 0 & 0 & 0 & 0 \\ 0 & \mathcal{V}^{2,1\dagger} & \mathcal{H}_1 + \hbar\omega_2 & \mathcal{V}^{2,1} & 0 & 0 & 0 \\ 0 & 0 & \mathcal{V}^{2,1\dagger} & \mathcal{H}_1 & \mathcal{V}^{2,1} & 0 & 0 \\ 0 & 0 & 0 & \mathcal{V}^{2,1\dagger} & \mathcal{H}_1 - \hbar\omega_2 & \mathcal{V}^{2,1} & 0 \\ 0 & 0 & 0 & 0 & \mathcal{V}^{2,1\dagger} & \mathcal{H}_1 - 2\hbar\omega_2 & \mathcal{V}^{2,1} \\ 0 & 0 & 0 & 0 & 0 & \mathcal{V}^{2,1\dagger} & \mathcal{H}_1 - 3\hbar\omega_2 \end{pmatrix}$$

with

$$\mathcal{V}^{2,1} = \begin{pmatrix} V^{2,1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & V^{2,1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & V^{2,1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & V^{2,1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & V^{2,1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & V^{2,1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & V^{2,1} \end{pmatrix}$$

C++ wrapper

The library includes wrappers to use with C++. The interfaces to the Fortran subroutines share the name of the aimed function with the appended particle `_C_`. The full set of wrappers are declared in the file `MultimodeFloquet.h`, which must be included in the C++ source code. As an example, we present a C++ program that evaluates the time-evolution of a driven qubit.

```
#include <iostream>
#include <complex>
#include <stdio.h>
#include <math.h>
#include <string.h>

using namespace std;
typedef std::complex<double> dcmplx;

#include "MultimodeFloquet.h"

extern "C" int h_floquet_size;

int main(){

    atom_c id;
    int info,N_;
    int jtotal;
    char name[] = "qubit";
    char manifold[] = "U";

    int r,m,l;
    int d_bare,total_frequencies;

    double t1,t2;
```



```

info = 0;
jtotal = 2;
floquetinit_c(&id,name,&info);

d_bare = id.d_bare;

dcmplx * U_AUX = new dcmplx [d_bare*d_bare];

int nm = 2;
int * modes_num = new int [nm];

modes_num[0] = 1;
modes_num[1] = 1;

total_frequencies = 0;
for(r=0;r<nm;r++){
    total_frequencies += modes_num[r];
}

mode_c * fields = new mode_c [total_frequencies];

// --- SET DRIVING PARAMETERS
fields[0].x = 0.0;
fields[0].y = 0.0;
fields[0].z = 2.0;
fields[0].phi_x = 0.0;
fields[0].phi_y = 0.0;
fields[0].phi_z = 0.0;
fields[0].omega = 0.0;
fields[0].N_Floquet = 0;

fields[1].x = 4.0;
fields[1].y = 0.0;
fields[1].z = 0.0;
fields[1].phi_x = 0.0;
fields[1].phi_y = 0.0;
fields[1].phi_z = 0.0;
fields[1].omega = 2.0;
fields[1].N_Floquet = 3;

// --- SET THE HAMILTONIAN COMPONENTS
sethamiltoniancomponents_c_
    (&id,&nm,&total_frequencies,modes_num,fields,&info);

// --- BUILD THE MULTIMODE FLOQUET MATRIX AND FIND ITS SPECTRUM
multimodefloquetmatrix_c_
    (&id,&nm,&total_frequencies,modes_num,fields,&info);

double * e_floquet = new double [h_floquet_size];
dcmplx * U_F = new dcmplx [h_floquet_size*h_floquet_size];

lapack_fulleigenvalues_c_
    (U_F,&h_floquet_size,e_floquet,&info);
// the diagonalization is done with the internal (Fortran)
// Hamiltonian (H_FLOQUET) and the calculated
// U_F is the transformation that diagonalise this Hamiltonian
// On the Fortran side, H_FLOQUET is deallocated after
// diagonalization.

//--- EVALUATE TIME-EVOLUTION OPERATOR IN THE BARE BASIS

N_ = 256;
t1 = 0.0;
for(r=1;r<N_;r++){
    t2 = r*100.0/N_;

```

```

    multimodetimeevolutionoperator_c_
        (&h_floquet_size,&nm,modes_num,U_F,e_floquet,&d_bare,fields,
         &t1,&t2,U_AUX,&info);
    for(l=0;l<d_bare*d_bare;l++) p_avg[l] = pow(abs(U_AUX[l]),2);
    write_matrix_c_(p_avg,&d_bare);
}
delete(e_floquet);
delete(U_F);
}

return 0;
}

```

Python wrapper

As an experimental feature, the library includes wrappers to use with Python through calls to the C++ wrapper functions using ctypes. The full set of wrapper functions are declared in the file `src/openmmf.py`, which must be included in working directory of the python script. As an example, we present a Python script that evaluates the time-evolution of a driven qubit. Other examples can be found in the folder `examples/PYTHON/`.

```

import numpy as np
import openmmf as openmmf
import matplotlib.pyplot as plt

# INITIALISE THE DATA TYPE
id = openmmf.atom_c_T()
name = 'qubit'
info = 0

openmmf.floquetinit(id,name,info=info)
d_bare = id.d_bare

# DEFINE THE NUMBER OF MODES
modes_num = np.array([1,1],dtype=np.int32)
nm = np.sum(modes_num)

# THIS INSTRUCTION DEFINES A TYPE OF
# ARRAY OF modes WITH nm COMPONENTS
fields = openmmf.mode_c_T*nm

# THIS INSTANCE DECLARES THE FIELDS
field = fields()

# DEFINE EACH ONE OF THE DRIVING FIELDS
field[0].x = 0,0
field[0].y = 0,0
field[0].z = 1,0
field[0].phi_x = 0
field[0].phi_y = 0
field[0].phi_z = 0
field[0].omega = 0
field[0].N_Floquet = 0

field[1].x = 2,0
field[1].y = 0,0
field[1].z = 0,0
field[1].phi_x = 0.0
field[1].phi_y = 0.0
field[1].phi_z = 0.0
field[1].omega = 1.0
field[1].N_Floquet = 8

N_ = 128
M_ = 128

```

```

# declare arrays to store the solution
P_AVG      = np.zeros([N_,d_bare*d_bare],dtype=np.double)
P_TimeEvol = np.zeros([N_,N_,d_bare*d_bare],dtype=np.double)

for m in range(N_):
    # SET A NEW FIELD CONFIGURATOIN, E.G.
    field[1].omega = 0.2 + m*2.0/N_

    # SET THE HAMILTONIAN COMPONENTS
    openmmf.sethamiltoniancomponents(id,modes_num,field,info)

    # BUILD THE MULTIMODE FLOQUET MATRIX
    h_floquet_size = openmmf.multimodefloquetmatrix(id,modes_num,field,info)

    # ALLOCATE ARRAYS FOR FLOQUET ENERGIES (E_FLOQUET), MICROMOTION (U_F),
    # TIME-AVERAGE TRANSITION PROBABILITY (p_avg), AND
    # AN AUXILIAR MATRIX (U_AUX)
    e_floquet = np.zeros(h_floquet_size,dtype=np.double)
    U_F       = np.zeros(h_floquet_size*h_floquet_size,dtype=np.complex)
    p_avg     = np.zeros(id.d_bare*id.d_bare,dtype=np.double)
    U_AUX     = np.zeros(id.d_bare*id.d_bare,dtype=np.complex)

    # DIAGONALISE THE MULTIMODE FLOQUET MATRIX
    openmmf.lapack_fulleigenvalues(U_F,h_floquet_size,e_floquet,info)

    # EVALUATE THE AVERAGE TRANSITION PROBABILITIES IN THE BARE BASIS
    openmmf.multimodetransitionavg(h_floquet_size,field,modes_num,
    U_F,e_floquet,d_bare,p_avg,info)
    P_AVG[m,:] = p_avg

    #SET THE INITIAL (T1) AND FINAL TIME (T2)
    t1 = 0.0
    for r in range(M_):
        t2 = r*32.0*4.0*np.arctan(1.0)/M_
        # EVALUATE THE TIME EVOLUTION OPERATOR
        openmmf.multimodetimeevolutionoperator(h_floquet_size,modes_num,U_F,e_floquet,d_bare,field,t1,t2,U_AUX,info)
        P_TimeEvol[m,r,:] = np.power(np.abs(U_AUX),2)

# DEALLOCATE ALL ARRAYS
# WARNING: THIS FUNCTION SHOULD BE CALLED TO CLEAR MEMORY
openmmf.deallocateall(id)

# PLOTTING
omega = np.linspace(0.2,2.2,N_)
#plotting the avg transition probability
plt.plot(omega,P_AVG[:,1])
plt.xlabel('frequency')
plt.ylabel('Average transition probability')
plt.show()

#%%
# plot the time evolution
t = np.linspace(0,32.0*4.0*np.arctan(1.0),M_)
X,Y = np.meshgrid(t,omega)
Z = P_TimeEvol[:, :, 1]

fig,ax = plt.subplots()
im = ax.pcolormesh(Y, X, Z,shading='auto')
ax.set_ylabel('time')
ax.set_xlabel('frequency')
fig.tight_layout()

```

References

- [1] Nakajima Shuta, Tomita Takafumi, Taie Shintaro, Ichinose Tomohiro, Ozawa Hideki, Wang Lei, et al. Topological Thouless pumping of ultracold fermions. *Nat Phys* 2016;12(4):296.
- [2] Harte TL, Bentine E, Luksch K, Barker AJ, Trypogeorgos D, Yuen B, et al. Ultracold atoms in multiple radio-frequency dressed adiabatic potentials. *Phys Rev A* 2018;97:013616.
- [3] Weidt S, Randall J, Webster SC, Lake K, Webb AE, Cohen I, et al. Trapped-ion quantum logic with global radiation fields. *Phys Rev Lett* 2016;117:220501.
- [4] Balasubramanian Gopalakrishnan, Neumann Philipp, Twitchen Daniel, Markham Matthew, Kolesov Roman, Mizuochi Norikazu, et al. Ultralong spin coherence time in isotopically engineered diamond. *Nat Mater* 2009;8(5):383.
- [5] Golter D Andrew, Baldwin Thomas K, Wang Hailin. Protecting a solid-state spin from decoherence using dressed spin states. *Phys Rev Lett* 2014;113(23):237601.
- [6] Zhou Brian B, Jerger Paul C, Shkolnikov Vladislav O, Heremans F Joseph, Burkard Guido, Awschalom David D. Holonomic quantum control by coherent optical excitation in diamond. *Phys Rev Lett* 2017;119(14):140503.
- [7] Tuorila Jani, Silveri Matti, Sillanpää Mika, Thuneberg Erkki, Makhlin Yuriy, Hakonen Pertti. Stark effect and generalized Bloch-Siegert shift in a strongly driven two-level system. *Phys Rev Lett* 2010;105(25):257003.
- [8] Yan Fei, Gustavsson Simon, Bylander Jonas, Jin Xiaoyue, Yoshihara Fumiki, Cory David G, et al. Rotating-frame relaxation as a noise spectrum analyser of a superconducting qubit undergoing driven evolution. *Nat Commun* 2013;4:2337.
- [9] Vion Denis, Aassime A, Cottet Audrey, Joyez PI, Pothier H, Urbina C, et al. Manipulating the quantum state of an electrical circuit. *Science* 2002;296(5569):886–9.
- [10] Baumgart I, Cai J-M, Retzker A, Plenio Martin B, Wunderlich Ch. Ultrasensitive magnetometer using a single atom. *Phys Rev Lett* 2016;116(24):240801.
- [11] Vandersypen Lieven MK, Chuang Isaac L. NMR techniques for quantum control and computation. *Rev Modern Phys* 2005;76(4):1037.
- [12] O'Brien Jeremy L, Furusawa Akira, Vučković Jelena. Photonic quantum technologies. *Nat Photonics* 2009;3(12):687.
- [13] Scully Marlan O, Zubairy M Suhail. *Quantum optics*. AAPT; 1999.
- [14] Sárkány L, Weiss P, Hattermann H, Fortágh J. Controlling the magnetic-field sensitivity of atomic-clock states by microwave dressing. *Phys Rev A* 2014;90(5):053416.
- [15] Kazakov Georgy A, Schumm Thorsten. Magic radio-frequency dressing for trapped atomic microwave clocks. *Phys Rev A* 2015;91(2):023404.
- [16] Lindner Netanel H, Refael Gil, Galitski Victor. Floquet topological insulator in semiconductor quantum wells. *Nat Phys* 2011;7(6):490.
- [17] Zhang J, Hess PW, Kyprianidis A, Becker P, Lee A, Smith J, et al. Observation of a discrete time crystal. *Nature* 2017;543(7644):217–20.
- [18] Mukherjee Seabrat, Rechtsman Mikael C. Observation of Floquet solitons in a topological bandgap. *Science* 2020;368(6493):856–9.
- [19] Gross Christian, Bloch Immanuel. Quantum simulations with ultracold atoms in optical lattices. *Science* 2017;357(6355):995–1001.
- [20] Ho Tak-San, Chu Shih-I, Tietz James V. Semiclassical many-mode Floquet theory. *Chem Phys Lett* 1983;96(4):464–71.
- [21] Shirley Jon H. Solution of the Schrödinger equation with a Hamiltonian periodic in time. *Phys Rev* 1965;138(4B):B979.
- [22] Song Yang, Kestner JP, Wang Xin, Sarma S Das. Fast control of semiconductor qubits beyond the rotating-wave approximation. *Phys Rev A* 2016;94(1):012321.
- [23] Weidner CA, Anderson Dana Z. Experimental demonstration of shaken-lattice interferometry. *Phys. Rev. Lett.* 2018;120:263201.
- [24] Price Hannah M, Zilberberg Oded, Ozawa Tomoki, Carusotto Iacopo, Goldman Nathan. Four-dimensional quantum Hall effect with ultracold atoms. *Phys Rev Lett* 2015;115(19):195303.
- [25] Martin Ivar, Refael Gil, Halperin Bertrand. Topological frequency conversion in strongly driven quantum systems. *Phys Rev X* 2017;7(4):041008.
- [26] Delande Dominique, Morales-Molina Luis, Sacha Krzysztof. Three-dimensional localized-delocalized Anderson transition in the time domain. *Phys Rev Lett* 2017;119(23):230404.
- [27] Goldman N, Budich JC, Zoller P. Topological quantum matter with ultracold gases in optical lattices. *Nat Phys* 2016;12(7):639.
- [28] Chu Shih-I. Recent developments in semiclassical Floquet theories for intense-field multiphoton processes. In: *Advances in atomic and molecular physics*, Vol. 21. Elsevier; 1985, p. 197–253.
- [29] Hemmerich Andreas. Effective time-independent description of optical lattices with periodic driving. *Phys Rev A* 2010;81:063626.
- [30] Verdeny Albert, Puig Joaquim, Mintert Florian. Quasi-periodically driven quantum systems. *Z Natforsch A* 2016;71(10):897–907.
- [31] Sinuco-Leon German. The openmmf library: open source software for multimode driven quantum systems. user manual. 2020, <https://github.com/openMMF/Multimode-Floquet>.
- [32] Sakurai Jun John, Commins Eugene D. *Modern quantum mechanics*. revised ed. American Association of Physics Teachers; 1995.
- [33] Anderson BE, Sosa-Martinez H, Rofio CA, Deutsch Ivan H, Jessen Poul S. Accurate and robust unitary transformations of a high-dimensional quantum system. *Phys Rev Lett* 2015;114:240401.
- [34] Randall J, Weidt S, Standing ED, Lake K, Webster SC, Murgia DF, et al. Efficient preparation and detection of microwave dressed-state qubits and qutrits with trapped ions. *Phys Rev A* 2015;91(1):012322.
- [35] Sinuco-Leon GA, Garraway BM, Mas H, Pandey S, Vasilakis G, Bopali V, et al. Microwave spectroscopy of radio-frequency-dressed ⁸⁷Rb. *Phys Rev A* 2019;100:053416.
- [36] Holthaus Martin. Floquet engineering with quasienergy bands of periodically driven optical lattices. *J Phys B: At Mol Opt Phys* 2015;49(1):013001.
- [37] Desbuquois Rémi, Messer Michael, Görg Frederik, Sandholzer Kilian, Jotzu Gregor, Esslinger Tilman. Controlling the Floquet state population and observing micromotion in a periodically driven two-body quantum system. *Phys Rev A* 2017;96(5):053602.
- [38] Lellouch Samuel, Bukov Marin, Demler Eugene, Goldman Nathan. Parametric instability rates in periodically driven band systems. *Phys Rev X* 2015;49(1):013001.
- [39] Anisimovas Egidijus, Žlabys Giedrius, Anderson Brandon M, Juzeliūnas Gediminas, Eckardt André. Role of real-space micromotion for bosonic and fermionic Floquet fractional Chern insulators. *Phys Rev B* 2015;91:245135.
- [40] MacQuarrie ER, Gosavi TA, Bhawe SA, Fuchs GD. Continuous dynamical decoupling of a single diamond nitrogen-vacancy center spin with a mechanical resonator. *Phys Rev B* 2015;92(22):224419.
- [41] Mas Hector, Pandey Saurabh, Vasilakis Georgios, von Klitzing Wolf. Bi-chromatic adiabatic shells for atom interferometry. *New J Phys* 2019;21(12):123039.
- [42] Sinuco-Leon GA. Decoherence-free radiofrequency dressed subspaces. 2020, arXiv:2007.15730.
- [43] Trypogeorgos D, Valdés-Curiel A, Lundblad N, Spielman IB. Synthetic clock transitions via continuous dynamical decoupling. *Phys Rev A* 2018;97:013407.
- [44] Golter D Andrew, Baldwin Thomas K, Wang Hailin. Protecting a solid-state spin from decoherence using dressed spin states. *Phys Rev Lett* 2014;113:237601.
- [45] Hirose Masashi, Aiello Clarice D, Cappellaro Paola. Continuous dynamical decoupling magnetometry. *Phys Rev A* 2012;86(6):062320.
- [46] Ma Wenchao, Zhou Longwen, Zhang Qi, Li Min, Cheng Chunyang, Geng Jianpei, et al. Experimental observation of a generalized thouless pump with a single spin. *Phys Rev Lett* 2018;120(12):120501.
- [47] Kitagawa Takuya, Berg Erez, Rudner Mark, Demler Eugene. Topological characterization of periodically driven quantum systems. *Phys Rev B* 2010;82:235114.
- [48] Khemani Védika, Lazarides Achilleas, Moessner Roderich, Sondhi SL. Phase structure of driven quantum systems. *Phys Rev Lett* 2016;116:250401.
- [49] Potirniche I-D, Potter AC, Schleier-Smith M, Vishwanath A, Yao NY. Floquet symmetry-protected topological phases in cold-atom systems. *Phys Rev Lett* 2017;119:123601.
- [50] Ji Kai, Fine Boris V. Suppression of heating in quantum spin clusters under periodic driving as a dynamic localization effect. *Phys Rev Lett* 2018;121:050602.
- [51] Yates Daniel, Lemonik Yonah, Mitra Aditi. Central charge of periodically driven critical Kitaev chains. *Phys Rev Lett* 2018;121:076802.