Numerical solution of the Lindblad master equation using the Runge-Kutta method implemented in Python

L. Hernández-Sánchez[a], I. A. Bocanegra-Garay[b], A. Flores-Rosas[c],
I. Ramos-Prieto[a], F. Soto-Eguibar[a], and H. M. Moya-Cessa[a]

[a]*Instituto Nacional de Astrofísica óptica y Electrónica,*
Calle Luis Enrique Erro No. 1, Santa María Tonantzintla, Puebla, 72840, Mexico.
[b]*Departamento de Física Teórica, Atómica y Óptica and Laboratory for Disruptive Interdisciplinary Science,*
Universidad de Valladolid, 47011 Valladolid, Spain
[c]*Facultad de Ciencias en Física y Matemáticas, Universidad Autónoma de Chiapas,*
Carretera Emiliano Zapata, Km. 8, Rancho San Francisco, 29050 Tuxtla Gutiérrez, Chiapas, Mexico.

The dynamics of open quantum systems is governed by the Lindblad master equation, which provides a consistent framework for incorporating environmental effects into the evolution of the system. Since exact solutions are rarely available, numerical methods become essential tools for analyzing such systems. This article presents a step-by-step implementation of the fourth-order Runge-Kutta method in `Python` to solve the Lindblad equation for a single quantized field mode subject to decay. A coherent state is used as the initial condition, and the time evolution of the average photon number is investigated. The proposed methodology enables transparent and customizable simulations of dissipative quantum dynamics, emphasizing a pedagogical approach that helps readers understand the numerical structure without relying on external libraries such as QuTiP. This standalone implementation offers full control over each integration step, making it particularly suitable for educational contexts and for exploring non-standard dynamics or introducing custom modifications to the Liouvillian.

1. Introduction

The study of open quantum systems is essential to understand how real systems interact with their environment. Unlike closed systems, whose evolution is purely coherent and reversible according to the Schrödinger equation, open systems undergo processes of dissipation and decoherence, which degrade quantum coherence due to coupling with an external environment. These effects are unavoidable in practice and are fundamental in areas such as quantum optics, quantum computing, quantum information, and spectroscopy [1-3].

The most general mathematical tool to describe the dynamics of such systems is the Lindblad master equation, which consistently models the time evolution of the density matrix by incorporating both coherent dynamics and irreversible effects induced by the environment [4,5]. Although this equation has a compact and elegant form, exact solutions are only possible in particular cases, due to the complexity of simultaneously treating both unitary and dissipative terms [6-8].

In this context, numerical methods have become indispensable tools for studying the evolution of open systems. Among these, the fourth-order Runge-Kutta (RK4) method stands out for its balance between accuracy, simplicity, and ease of implementation to solve ordinary differential equations, making it an excellent option for approximating solutions of the Lindblad equation in increasingly complex systems [9-10].

In previous studies focused on closed systems, the effectiveness of the RK4 method in simulating the dynamics of radiation-matter interaction models governed by the Schrödinger equation has been demonstrated [11]. In the present work, this strategy is extended to open quantum systems by presenting a detailed implementation of the RK4 method to numerically solve the Lindblad master equation.

As an illustrative example, the case of a single quantized field mode subject to decay is analyzed, evolving from an initial coherent state. The implementation of this simulation in `Python` is presented, including the computation of the average photon number as a physical observable and its comparison with the exact analytical solution, thereby validating the accuracy of the proposed approach.

Beyond its numerical utility, the methodology is developed with a pedagogical emphasis, enabling readers to grasp the structure of the algorithm and the role of each term in the evolution equation without relying on specialized packages such as QuTiP. This educational perspective also provides the flexibility needed to extend the method to more general or nonstandard dissipative scenarios, such as time-dependent Hamiltonians or state-dependent dissipation.

The main objective of this work is to provide a clear and accessible guide to simulate the dynamics of open quantum systems using a basic computational approach, promoting the understanding of these tools in both educational and research contexts.

The structure of the article is as follows. Section 2 presents the physical model and its analytical solution. Section 3 describes step-by-step the implementation of the RK4 method to solve the Lindblad equation. In Sec. 4, numerical results are compared with the analytical solution. Finally, Section 5 presents the general conclusions of the work.

2. The physical model and its analytical solution

This section describes the physical system analyzed throughout the article: a single mode of a quantized electromagnetic field confined in a lossy optical cavity. This system can be interpreted as a quantum harmonic oscillator that loses energy due to its interaction with the environment [12].

Figure 1 shows a schematic representation of the considered system. A single field mode with frequency ω_c is confined between two semitransparent mirrors. The interaction with the environment causes photons to leak out of the cavity, modeled through a decay process characterized by the rate γ.

The time evolution of the quantum state of the system, described by the density matrix $\hat{\rho}(t)$, is governed by the Lindblad master equation. For this setup, the free Hamiltonian is:

$$\hat{H} = \omega_c \hat{a}^\dagger \hat{a}, \tag{1}$$

where $\hat{a}$ and $\hat{a}^\dagger$ are the annihilation and creation operators of the field mode, and ω_c is its frequency.

The coupling to the environment is modeled using a single collapse operator:

$$\hat{C} = \sqrt{\gamma}\hat{a}, \tag{2}$$

where γ represents the photon loss rate.

The Lindblad master equation for this system takes the form [1-3]:

$$\frac{d\hat{\rho}}{dt} = -i[\hat{H}, \hat{\rho}] + \gamma \left(\hat{a}\hat{\rho}\hat{a}^\dagger - \frac{1}{2}\left\{ \hat{a}^\dagger\hat{a}, \hat{\rho} \right\} \right). \tag{3}$$

Each term in this equation has a specific role:

- The commutator $-i[\hat{H}, \hat{\rho}]$ represents the unitary evolution of the system, analogous to the Schrödinger equation but applied to the density matrix.

FIGURE 1. Schematic representation of a single quantized field mode with frequency ω_c, confined in an optical cavity with losses characterized by decay rate γ.

- The term $\hat{a}\hat{\rho}\hat{a}^\dagger$ describes quantum jump processes, in which the system emits a photon.

- The anticommutator terms $-(1/2)\{\hat{a}^\dagger\hat{a}, \hat{\rho}\}$ account for the gradual loss of coherence due to environmental coupling.

In this work, the system is initialized in a coherent state $|\alpha\rangle$, which is an eigenstate of the annihilation operator: $\hat{a}|\alpha\rangle = \alpha|\alpha\rangle$. This type of state describes a quantum field whose statistical properties resemble those of a classical wave [3].

Although the Lindblad equation generally does not admit analytical solutions, this particular case, a single-field mode subject to decay, does have a known solution. As shown in Chapter 5 of [13], a coherent state remains coherent throughout the evolution, with an amplitude that decays exponentially in time. The density matrix at time t is:

$$\hat{\rho}(t) = |\alpha(t)\rangle\langle\alpha(t)|, \tag{4}$$

where the amplitude evolves as:

$$\alpha(t) = \alpha\, e^{-\frac{\gamma}{2}t}\, e^{-i\omega_c t}. \tag{5}$$

A physically relevant observable characterizing the system dynamics is the average number of photons. In the density matrix formalism, it is computed as:

$$\langle\hat{n}\rangle(t) = \mathrm{Tr}\left[\hat{a}^\dagger\hat{a}\,\hat{\rho}(t)\right]. \tag{6}$$

Since $\hat{\rho}(t)$ remains a pure coherent state and $\hat{a}|\alpha(t)\rangle = \alpha(t)|\alpha(t)\rangle$, it follows that:

$$\langle\hat{n}\rangle(t) = \langle\alpha(t)|\hat{a}^\dagger\hat{a}|\alpha(t)\rangle = |\alpha(t)|^2 = |\alpha|^2 e^{-\gamma t}. \tag{7}$$

This expression shows that the average photon number decays exponentially at a rate γ, reflecting the energy loss of the field due to its coupling to the environment.

For simplicity, this work considers a zero temperature environment, which allows us to focus on numerical implementation without additional thermal excitation effects. In this regime, the system evolves towards a stationary state as $t \to \infty$, where the density matrix tends to $\hat{\rho}_{\mathrm{ss}} = |0\rangle\langle 0|$. This vacuum state represents the complete decay of the field energy, and the exact steady-state value of the average photon number is $\langle\hat{n}\rangle_{\mathrm{ss}} = 0$.

However, the model can be naturally extended to finite-temperature environments by incorporating additional dissipative terms that depend on the mean thermal photon number $\bar{n}$. In such cases, the system asymptotically approaches a thermal mixed state instead of the vacuum, and the steady-state average photon number becomes $\langle\hat{n}\rangle_{\mathrm{ss}} = \bar{n}$. This thermalization behavior, along with the uniqueness and structure of the steady state, can be directly understood from the spectral properties of the Liouvillian superoperator [4].
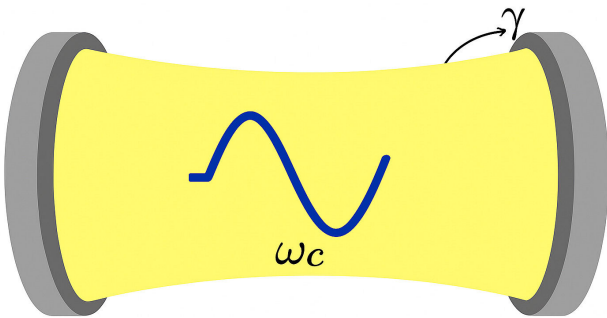
3. Methodology

This section provides a detailed description of the application of the RK4 method to solve the Lindblad master Eq. (3) for the case of a single quantized field mode subject to decay. This approach not only offers high accuracy and ease of implementation, but can also be naturally extended to more complex models, such as systems with multiple interacting modes, coupling to thermal reservoirs, or several dissipation channels, situations in which analytical solutions are generally unavailable.

3.1. Problem formulation

As discussed previously, the dynamics of the system is fully determined by the time evolution of its density matrix $\hat{\rho}(t)$, which is obtained by solving the Lindblad master Eq. (3). In its general form, this equation can be written as:

$$\frac{d\hat{\rho}}{dt} = \mathcal{L}[\hat{\rho}], \tag{8}$$

where $\mathcal{L}$ denotes the *Lindblad superoperator*, which encapsulates both the coherent dynamics of the system and the dissipative effects arising from its interaction with the environment.

Formally, the solution to this equation can be expressed as:

$$\hat{\rho}(t) = e^{\mathcal{L}t}\hat{\rho}(0), \tag{9}$$

where $\hat{\rho}(0) = |\psi(0)\rangle\langle\psi(0)|$ represents the initial condition of the system.

The main objective of this section is to develop, step-by-step, a methodology based on the RK4 algorithm and to implement it in the `Python` programming language, in order to approximate the solution of Eq. (8). As an illustrative example, the model of a single-field mode undergoing decay is considered. The analytical solution previously presented allows for direct validation of the numerical results.

It is important to note that this methodology is not limited to the specific case analyzed here. It can be readily generalized to more complex quantum systems, including multiple field modes, interaction with atoms, coupling to thermal environments, or arbitrary initial conditions.

3.2. Fourth-order Runge-Kutta method adapted to matrices

The RK4 method is a widely used numerical technique for solving ordinary differential equations (ODE) of the form [9]:

$$\frac{dy}{dt} = f(t, y), \quad y(0) = y_0. \tag{10}$$

The core idea of RK4 is to approximate the solution using a weighted combination of several evaluations of the function $f(t, y)$ within each finite time increment Δt, which represents the time step used in the numerical integration. The update scheme is expressed as:

$$t_{i+1} = t_i + \Delta t, \tag{11}$$

$$k_1 = f(t_i, y_i), \tag{12}$$

$$k_2 = f\left(t_i + \frac{\Delta t}{2}, y_i + \frac{\Delta t}{2}k_1\right), \tag{13}$$

$$k_3 = f\left(t_i + \frac{\Delta t}{2}, y_i + \frac{\Delta t}{2}k_2\right), \tag{14}$$

$$k_4 = f\left(t_i + \Delta t, y_i + \Delta t\, k_3\right), \tag{15}$$

$$y_{i+1} = y_i + \frac{\Delta t}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right). \tag{16}$$

Its main advantage lies in its fourth-order accuracy, which means that the local error per time step is on the order of $\mathcal{O}(\Delta t^5)$, while the total cumulative error is $\mathcal{O}(\Delta t^4)$. Although this method requires four evaluations of the function f per time step, RK4 offers an excellent balance between accuracy and computational efficiency.

In the context of the Lindblad master equation, the function $f(t, y)$ is replaced by the action of the Lindblad superoperator $\mathcal{L}$ on the density matrix:

$$f(t, \hat{\rho}) = \mathcal{L}[\hat{\rho}], \tag{17}$$

where $\hat{\rho}(t)$ is a Hermitian matrix that describes the quantum state of the system. The RK4 scheme can be applied without formal modification, although the operations must now be interpreted as acting on complex matrices rather than on scalars or vectors. In this case, the variable y is identified with the density matrix $\hat{\rho}$, and the quantities k_1, k_2, k_3, and k_4 are also matrices of the same dimension, calculated iteratively by the action of the superoperator $\mathcal{L}$.

One of the principal advantages of this method is its ease of implementation in programming languages such as `Python`, making it an attractive tool for studying the dynamics of open quantum systems. Although its computational cost may be higher than that of simpler algorithms, such as the Euler method, its accuracy and numerical stability make it especially useful for simulating systems without analytical solutions where reliable numerical integration is required.

It is worth noting that this implementation employs a fixed time step Δt, which helps to clearly illustrate the structure of the algorithm and ensures consistency across different numerical runs. This choice aligns with the pedagogical purpose of this work, which is to offer a transparent and accessible introduction to the numerical integration of the Lindblad master equation.

However, in scenarios where the system exhibits fast transients or widely separated time scales–such as strong dissipation regimes or complex interactions–using a fixed step size may lead to either poor accuracy or inefficient use of computational resources. In such cases, it is advisable to employ adaptive step-size techniques that dynamically adjust Δt based on an estimate of the local truncation error.

A well-established solution is the Runge-Kutta-Fehlberg (RKF) method, which combines two embedded Runge-Kutta

formulas of different orders (typically fourth and fifth) to control the error and adapt the time step accordingly [14-16]. The methodology presented in this work can be naturally extended to incorporate such schemes, offering higher numerical efficiency without altering the underlying structure of the integration algorithm.

3.3. Implementation in `python`

This subsection presents the implementation of the RK4 method in `Python` to numerically solve the Lindblad master Eq. (8), applied to the model of a single quantized field mode subject to decay. The code developed here applies the RK4 iterative scheme directly in the matrix context, meaning that the variables involved are treated as operators (*i.e.*, dense complex matrices) rather than scalar functions. The time evolution is computed on a truncated Fock basis of finite dimension, which is appropriate for representing initial coherent states.

The following code illustrates the implementation of this methodology step-by-step.

1. **Library importation**

    The first step in the implementation consists of importing the fundamental libraries that allow for performing numerical operations, manipulating matrix structures, and generating graphical representations of the results. These tools are essential for defining the operators of the quantum system, computing the time evolution using the RK4 method, and visualizing the dynamic behavior of the system.

    ```python
    import numpy as np
    import matplotlib.pyplot as plt
    import math
    from scipy.linalg import null_space
    ```

    - `numpy`: This is the standard `Python` library for numerical calculations with multidimensional arrays. In this work, it is used to define the system operators, manipulate the density matrix, and carry out matrix multiplications and summations required for the time evolution.

    - `matplotlib.pyplot`: This library makes it possible to create two-dimensional plots. In the context of quantum simulation, it is used to visualize the time evolution of physical observables, thereby facilitating the interpretation of the results.

    - `math`: This library provides standard mathematical functions. In particular, it is used to compute exact integer factorials, which are necessary for constructing the initial coherent state as a superposition of Fock states.

    - `scipy.linalg.null_space`: This function is used to compute the steady state of the system

by finding the null space of the Liouvillian superoperator. It allows for direct numerical access to the long-time limit of the dynamics, complementing the RK4 simulation.

2. **Definition of the system parameters**

    In this step, the fundamental parameters of the model are defined. These parameters describe the physical properties and initial conditions of the quantum system under study.

    ```python
    N       = 30
    omega_c = 0.9
    gamma   = 0.3
    alpha   = 3.0
    t_f     = 20
    dt      = 0.01
    t_list  = np.arange(0, t_f, dt)
    ```

    - `N`: Specifies the dimension of the truncated Hilbert space, which corresponds to the maximum number of Fock states used to represent the field.

    - `omega_c`: Sets the angular frequency of the single quantized mode of the electromagnetic field confined in the optical cavity.

    - `gamma`: Represents the decay rate of the cavity, which models the photon loss due to coupling with the external environment.

    - `alpha`: Determines the complex amplitude of the initial coherent state $|\alpha\rangle$ used to initialize the system.

    - `t_f`: Specifies the total duration of the simulation. The evolution of the system will be computed from $t = 0$ to $t = t_f$.

    - `dt`: Defines the size of each discrete time step used by the Runge-Kutta integration algorithm. Smaller values improve accuracy but require more computational resources.

    - `t_list`: Creates a NumPy array of equally spaced time points using the function `np.arange`. This array defines the specific time instants at which the observables of the system will be evaluated during the simulation.

3. **Definition of operators and system Hamiltonian**

    This block defines the fundamental operators that describe the quantum dynamics of a single field mode confined in a cavity. These operators include the annihilation operator $\hat{a}$, the creation operator $\hat{a}^\dagger$, and the number operator $\hat{n} = \hat{a}^\dagger \hat{a}$. From these, the free Hamiltonian of the system is constructed, as well as the collapse operator that accounts for the dissipative interaction of the system with the environment.

```
1  def destroy(N):
2      return np.diag(np.sqrt(np.arange(1, N)),
            1)
3  a = destroy(N)
4  ad = a.conj().T
5  n_op = ad @ a
6  H = omega_c * n_op
7  c = np.sqrt(gamma) * a
```

- destroy(N): This function returns the annihilation operator $\hat{a}$ represented as an $N \times N$ matrix in the truncated Fock basis.

- a: Represents the annihilation operator $\hat{a}$, which lowers the photon number of the field mode by one.

- ad: Represents the creation operator $\hat{a}^\dagger$, obtained as the complex conjugate transpose of the annihilation operator. It increases the photon number by one when acting on a Fock state.

- n_op: Defines the number operator $\hat{n} = \hat{a}^\dagger \hat{a}$ using the matrix product ad @ a. In Python, the symbol @ is used to perform matrix multiplication, unlike *, which denotes element-wise multiplication.

- H: Constructs the Hamiltonian $\hat{H} = \omega_c \hat{n}$, which describes the unitary evolution of the quantized field mode with frequency ω_c in the absence of interaction with the environment.

- c: Defines the collapse operator $\hat{C} = \sqrt{\gamma}\hat{a}$, which characterizes the photon loss process due to coupling with the external environment.

4. **Definition of the Lindblad master equation**

In this step, the Lindblad master equation is formulated to describe the time evolution of the density matrix $\hat{\rho}(t)$ in an open quantum system. This equation incorporates both the coherent dynamics governed by the system Hamiltonian and the dissipative effects arising from its interaction with the environment.

```
1  def lindblad_rhs(rho, H, c_ops):
2      commutator = -1j * (H @ rho - rho @ H)
3      dissipator = np.zeros_like(rho, dtype=
            complex)
4      for c in c_ops:
5          dissipator += c @ rho @ c.conj().T -
                0.5 * (c.conj().T @ c @ rho + rho
                @ c.conj().T @ c)
6      return commutator + dissipator
```

- lindblad_rhs(rho, H, c_ops): This function computes the right-hand side of the Lindblad master equation.

- rho: Represents the density matrix of the system at a given time.

- H: The system Hamiltonian, which generates the unitary part of the evolution.

- c_ops: A list of collapse operators $\hat{C}_n$ that model dissipation channels. In this particular model, only one collapse operator is used: $\hat{C} = \sqrt{\gamma}\hat{a}$.

- commutator: Computes the coherent term $-i[\hat{H}, \hat{\rho}]$, corresponding to unitary evolution.

- dissipator: Computes the dissipative part of the Lindblad equation, which accounts for the non-unitary evolution due to the interaction with the environment.

5. **Implementation of the RK4 method**

In this step, the RK4 method is applied to numerically solve the Lindblad master equation. This technique approximates the time evolution of the density matrix $\hat{\rho}(t)$ by dividing the time domain into discrete steps and computing the solution iteratively.

```
1  def rk4_step(rho, H, c_ops, dt):
2      k1 = lindblad_rhs(rho, H, c_ops)
3      k2 = lindblad_rhs(rho + dt/2 * k1, H,
            c_ops)
4      k3 = lindblad_rhs(rho + dt/2 * k2, H,
            c_ops)
5      k4 = lindblad_rhs(rho + dt * k3, H, c_ops
            )
6      return rho + (dt/6) * (k1 + 2*k2 + 2*k3 +
            k4)
```

- rk4_step: This function performs a single integration step using the RK4 method applied to the Lindblad equation. It takes as input the current density matrix rho, the system Hamiltonian H, the list of collapse operators c_ops, and the time step size dt.

- k1, k2, k3, k4: These are the four intermediate evaluations required by the RK4 scheme to estimate the derivative of the density matrix. Each one corresponds to a different point in time and uses a different approximation of the state.

- Each k is computed by calling the lindblad_rhs function, which returns the value of the time derivative at the corresponding state.

- The weighted combination of the four terms (k1 + 2*k2 + 2*k3 + k4) produces a fourth-order accurate estimate of the state at the next time step.

- The final result is the updated density matrix at the next time step, computed with high precision and numerical stability.

6. **Definition of the initial state of the system**

In this step, the initial condition of the system is established. As the initial state, a coherent state is selected. Instead of representing this state as a pure state vector, its corresponding density matrix $\hat{\rho}_0 = |\alpha\rangle\langle\alpha|$ is

used, which is necessary for solving the Lindblad master equation.

```
1  def coherent_dm(N, alpha):
2      vec = np.array([alpha**m / math.sqrt(math
           .factorial(m)) for m in range(N)],
           dtype=complex)
3      vec *= np.exp(-0.5 * abs(alpha)**2)
4      return np.outer(vec, vec.conj())
5  rho_0 = coherent_dm(N, alpha)
```

- – coherent_dm(N, alpha): This function constructs the density matrix of the coherent state $|\alpha\rangle$ in a Fock basis truncated to dimension N.

- – np.outer(vec, vec.conj()): Computes the outer product between a complex vector $|\psi\rangle$ and its conjugate transpose $\langle\psi|$, resulting in a pure state density matrix. In this context, the use of np.outer ensures the construction of a square matrix of size $N \times N$ that accurately represents the initial field state.

- – rho_0: Denotes the initial density matrix of the system, which is used as the input for the evolution governed by the Lindblad master equation.

7. **Time evolution of the density matrix and calculation of observables**

Once the time evolution of the density matrix $\hat{\rho}(t)$ has been obtained, it is possible to calculate any physical observable of the system. In this work, the mean photon number $\langle\hat{n}(t)\rangle$ is evaluated, which provides a direct measure of the energy stored in the field mode over time.

```
1  n_expect = []
2  for t in t_list:
3      n_expect.append(np.real(np.trace(n_op @
           rho_0)))
4      rho_0 = rk4_step(rho_0, H, [c], dt)
```

- – n_expect: A list that stores the mean photon number $\langle\hat{n}(t)\rangle$ at each time step.

- – rho_0: Represents the density matrix that is updated at each time step using the function rk4_step, previously defined. In each iteration, it is replaced by the matrix at the next time step.

- – np.trace(n_op @ rho_0): Computes the expectation value of the number operator in the current state using the formula $\langle\hat{n}\rangle = \text{Tr}(\hat{n}\hat{\rho})$. The result is converted to its real part using np.real() to eliminate possible small imaginary components arising from numerical rounding.

3.4. Method validation

To verify the precision of the numerical implementation of the RK4 method applied to the Lindblad master equation,

we compare the results for a physical observable with its corresponding analytical solution. Specifically, we analyze the time evolution of the average photon number $\langle\hat{n}(t)\rangle$ in a single-mode quantum field initially prepared in a coherent state $|\alpha\rangle$ and subject to losses characterized by a decay rate γ. The exact analytical solution for this case is given by Eq. (7).

To carry out the validation, we use the analytical expression for the mean photon number $\langle\hat{n}(t)\rangle$, derived at the end of Sec. 2, where both the transient and long-time behaviors were discussed. This solution provides a direct benchmark for the numerical results obtained with the RK4 method and serves as a reference for the asymptotic behavior of the system at zero temperature.

```
1  n_analytic = np.abs(alpha)**2 * np.exp(-gamma *
       t_list)
2  n_ss_analytic = 0.0
```

To confirm that the numerical method correctly captures the steady-state behavior of the open quantum system, the Liouvillian superoperator is constructed explicitly in matrix form and its null space is computed. The steady state corresponds to the eigenvector associated with the zero eigenvalue of this operator.

```
1  I = np.eye(N)
2  L = -1j * (np.kron(I, H) - np.kron(H.T, I))
3  L += np.kron(c, c.conj()) - 0.5 * (np.kron(I, c.
       conj().T @ c) + np.kron((c.conj().T @ c).T, I)
       )
4  L = L.reshape(N*N, N*N)
5
6  nullvec = null_space(L)
7  if nullvec.shape[1] == 0:
8      raise ValueError("No steady state found (null
           space is empty).")
9
10 rho_ss_vec = nullvec[:, 0]
11 rho_ss = rho_ss_vec.reshape((N, N))
12 rho_ss = rho_ss / np.trace(rho_ss)
13
14 n_ss_liouvillian = np.real(np.trace(n_op @ rho_ss)
       )
```

- **n_analytic**: Computes the analytical expression for the average photon number in a decaying coherent state.

- **n_ss_analytic**: Stores the expected steady-state value $\langle\hat{n}\rangle$ at zero temperature, which corresponds to the vacuum.

- **L**: Constructs the Liouvillian superoperator in vectorized form using Kronecker products to represent the Lindblad master equation as a matrix equation.

- **null_space(L)**: Computes the null space of the Liouvillian matrix. A non-trivial eigenvector (if found) corresponds to the steady state ρ_{ss}.

- **rho_ss**: The steady-state density matrix, reshaped and normalized.

- **n_ss_liouvillian**: Expectation value of the number operator evaluated in the steady state obtained from the Liouvillian formalism.

The final comparison of results is shown graphically:

```python
plt.figure(figsize=(8, 5))
plt.plot(t_list, n_expect, label="RK4_(numerical)"
    , linewidth=2.5, color='orange')
plt.plot(t_list, n_analytic, '--', label="
    Analytical_solution", linewidth=1.5, color='
    blue')
plt.axhline(y=n_ss_liouvillian, color='skyblue',
    linestyle='-', linewidth=2.5,
            label="Steady_state_(Liouvillian)")
plt.axhline(y=n_ss_analytic, color='green',
    linestyle=':', linewidth=1.5,
            label="Analytical_steady_state")
plt.xlabel(r"Time_$t$", fontsize=14)
plt.ylabel(r"$\langle_\hat{n}(t)_\rangle$",
    fontsize=14)
plt.title("Mean_photon_number:_evolution_and_
    steady_state", fontsize=14)
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.savefig("Average_photon_number.pdf", format="
    pdf")
plt.show()
```

- `plt.plot(t_list, n_expect, ...)`: Plots the numerical values of the average photon number obtained using the RK4 method (orange line).

- `plt.plot(t_list, n_analytic, ...)`: Plots the exact analytical solution as a dashed blue line.

- `plt.axhline(..., n_ss_liouvillian, ...)`: Draws a horizontal line showing the steady-state value from the Liouvillian approach (sky blue).

- `plt.axhline(..., n_ss_analytic, ...)`: Draws a horizontal line indicating the analytically expected steady-state value (green dotted line).

- The graph offers a visual validation of the method: the numerical and analytical curves overlap during the transient regime and converge to the same steady-state value.

The resulting graph is shown in Fig. 2.

4. Results and discussion

The numerical implementation of the RK4 method to solve the Lindblad master equation exhibits excellent agreement with the exact analytical solution for the average photon number in a decaying coherent field mode. This correspondence, shown in Fig. 2, validates both the precision and the numerical stability of the proposed scheme.

The numerical results accurately reproduce the exponential decay law predicted by theory for the time evolution of the average number of photons. The trajectory obtained using RK4 remains close to the analytical solution throughout the simulated interval, without displaying significant cumulative deviations.
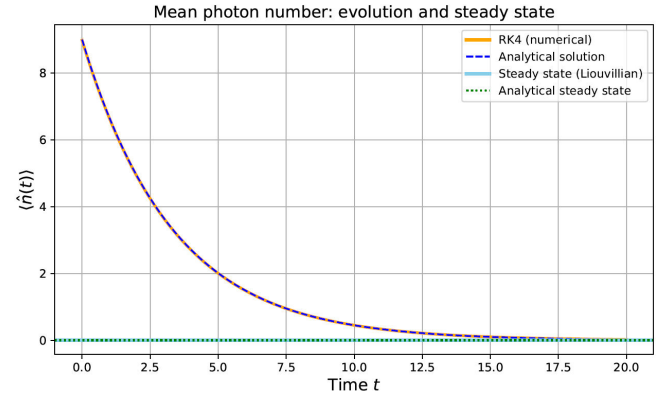

FIGURE 2. Time evolution of the average photon number $\langle \hat{n}(t) \rangle$ for a single-mode decaying quantum field. The solid orange line corresponds to the numerical solution obtained using the RK4 method, the dashed blue line represents the exact analytical result, and the horizontal lines indicate the steady-state predictions from both approaches.

In addition to verifying the transient behavior, the implementation also recovers the expected steady state by computing the null space of the Liouvillian superoperator. This numerical extraction of the stationary state confirms that the method captures both short- and long-term dynamics. The ability to access the steady state in matrix form further enriches the applicability of the code for the analysis of equilibrium and spectral studies of open quantum systems.

Although existing high-level libraries such as `QuTiP` provide efficient and user-friendly solvers for quantum master equations, they encapsulate the numerical procedure behind abstraction layers. In contrast, the RK4 implementation developed here explicitly reveals each step of the time evolution, making it particularly valuable for educational purposes. This clarity helps learners to understand how dissipation enters the dynamics and how the Lindblad structure is numerically realized.

By implementing the algorithm from scratch, users gain full control over the numerical evolution, enabling them to explore modified Hamiltonians, time-dependent interactions, or nonstandard dissipative terms-scenarios that may not be directly accessible with prebuilt solvers.

Thus, the proposed approach is not intended to complement libraries like `QuTiP`, but rather to reveal and demystify the internal workings that such libraries typically conceal. This makes it especially suitable for pedagogical contexts and for researchers looking for a transparent and adaptable numerical scheme.

5. Conclusions

This work presented a step-by-step implementation of the RK4 method to numerically solve the Lindblad master equation, applied to the case of a single coherent field mode subject to decay. The approach emphasizes clarity and pedagogical accessibility, providing an explicit numerical routine

suitable for students and researchers beginning in the study of open quantum systems.

The methodology proved capable of reproducing not only the expected transient dynamics of the average photon number, but also its steady-state behavior, confirming the internal consistency and physical precision of the numerical scheme. The explicit recovery of the steady state from the Liouvillian spectrum also offers a meaningful tool for future studies involving thermalization, equilibrium, or non-Hermitian effects.

By modifying only the Hamiltonian and collapse operators, the same framework can be applied to a wide variety of dissipative systems, including multimode fields, atom-field interactions, or thermal reservoirs.

Overall, this implementation provides a solid foundation for future explorations, combining physical transparency, computational control, and educational value within a reproducible open source environment. It is particularly useful in contexts where understanding the underlying structure of the evolution equation is essential, offering an open alternative to high-level packages that obscure the numerical treatment of dissipation.

Acknowledgments

1. H.-P. Breuer and F. Petruccione, The theory of open quantum systems (Oxford University Press on Demand, 2002).

2. H. J. Carmichael, An open systems approach to quantum optics (Springer-Verlag, 1993). `https://doi.org/10.1007/978-3-540-47620-7`.

3. C. Gerry and P. Knight, Introductory Quantum Optics (Cambridge University Press, 2004), `https://doi.org/10.1017/CBO9780511791239`

4. L. M. Arévalo-Aguilar and H. Moya-Cessa, Cavidad con pérdidas: una descripción usando superoperadores, *Rev. Mex. Fis.* **42** (1995) 675.

5. D. Manzano, A Short Introduction to the Lindblad Master Equation, *AIP Advances* **10** (2020) 025106, `https://doi.org/10.1063/1.5115323`

6. L. Hernández-Sánchez *et al.*, Exact solution for the interaction of two decaying quantized fields, *Opt. Lett.* **48** (2023) 5435, `https://doi.org/10.1364/OL.503837`

7. L. Hernández-Sánchez *et al.*, Exact solution of the master equation for interacting quantized fields at finite temperature decay (2024), `https://arxiv.org/abs/2410.08428`

8. L. Hernández-Sánchez et al., Dynamics of the Interaction Between Two Coherent States in a Cavity with Finite Temperature Decay, *Dynamics* **5** (2025) 4, `https://doi.org/10.3390/dynamics5010004`

9. J. C. Butcher, Numerical Methods for Ordinary Differential Equations (John Wiley & Sons, Ltd, 2016), `https://onlinelibrary.wiley.com/doi/book/10.1002/9781119121534`

10. W. H. Press, B. P. Flannery, and S. A. Teukolsky, Numerical Recipes: The Art of Scientific Computing 1st ed. (Cambridge University Press, 1986),

11. L. Hernández-Sánchez *et al.*, Formas de Línea Atómicas en Modelos de Tipo Jaynes-Cummings (Editorial Académica Española, 2023), p. 150.

12. W. H. Louisell, Quantum Statistical Properties of Radiation (John Wiley & Sons, Inc., New York, 1990), `https://ui.adsabs.harvard.edu/abs/1990qspr.book...`

13. H. M. Moya-Cessa and F. Soto-Eguibar, Introduction To Quantum Optics (Rinton Press, 2011), `https://www.researchgate.net/publication/259582579_INTRODUCTION_TO_QUANTUM_OPTICS`

14. E. Fehlberg, Low-order classical Runge-Kutta formulas with step size control and their application to some heat transfer problems, NASA Technical Report NASA-TR-R-315, NASA Marshall Space Flight Center, Huntsville, AL, United States (1969), `https://ntrs.nasa.gov/citations/19690021375`, Work of the US Gov. Public Use Permitted.

15. P. Hammachukiattikul, *et al.*, Runge-Kutta Fehlberg Method for Solving Linear and Nonlinear Fuzzy Fredholm Integro-Differential Equations, *Appl. Math. Inf. Sci.* **15** (2021) 43, `https://doi.org/10.18576/amis/150106`

16. C. Nwankwo and W. Dai, An adaptive and explicit fourth order Runge-Kutta-Fehlberg method coupled with compact finite differencing for pricing American put options, *Japan J. Indust. Appl. Math.* **38** (2021) 921, `https://doi.org/10.1007/s13160-021-00470-2`