



---

---

## Two-qubit CZ gate implementation with trapped neutral atoms: a numerical simulation

Alice Pagano Mat. 1236916 [alice.pagano@studenti.unipd.it](mailto:alice.pagano@studenti.unipd.it)  
Michele Puppin Mat. 1227474 [michele.puppin@studenti.unipd.it](mailto:michele.puppin@studenti.unipd.it)

(Dated: March 2, 2021)

Multi-qubit gates can be implemented with trapped neutral atoms by driving them to highly excited Rydberg states, in which nearby Rydberg atoms can interact very strongly and in a controlled way. In this Report, we simulate the dynamics of a system under the application of a controlled-phase gate physically implemented in [1]. After a physical and theoretical description of the two-qubit CZ gate design, the dynamics of a two-qubit system is simulated in both the perfect and imperfect Rydberg blockade regime by varying the optimal parameters suggested. Then, we consider the CZ gate in the case of a chain of  $N$ -qubit. A Gaussian noise is introduced on the optimal parameters to investigate noise effects with an increasing number of qubits. Numerical methods for performing unitary time evolution are also discussed and their performances are tested.

---

### CONTENTS

I.	Introduction	1
II.	Physical implementation of single-qubit and two-qubit gates	2
III.	Theoretical design of two-qubit CZ gate	5
IV.	Code implementation of two-qubit CZ gate	7
1.	Two-qubit system	7
2.	Chain of N-qubit	8
V.	Numerical methods for time-dependent Schrödinger equation solvers	9
1.	Spectral method	9
2.	Crank-Nicolson method	10
3.	Crank-Nicolson method with LU decomposition	10
VI.	Results	12
1.	CZ gate in a two-qubit system	12
A.	Correct behaviour check and optimal parameters variation	12
B.	Measurement process with Gaussian noise on $\tau$ and $\Delta$	15
2.	CZ gate in a chain of N-qubit	16
3.	Timing analysis of time evolution methods	17
VII.	Conclusions	19
	References	19

---

---

## I. INTRODUCTION

Nowadays, many research groups and companies are investing time and resources for the development of practical quantum computers. The basic idea of a quantum computer is much the same of a classical one. The information is stored in a register of *qubits*, namely quantum mechanical systems with two orthogonal states conventionally denoted by  $|0\rangle$  and  $|1\rangle$ . The processing tasks are carried out by *quantum logic gates*. These quantum logic gates are connected together in a *quantum circuit* in order to carry out specified processing tasks [2]. It has not yet been decided which basic physical principle is the most suitable for physical implementations of qubits and in general the most important functional parameters of a quantum computer. According to current knowledge, the most promising physical systems for building a next-generation quantum computer are superconductors, atomic traps and ion traps.

In recent years, quantum operations with *trapped neutral atoms* have made significant progress and have reached values in the range of those of ion traps and superconductors. Indeed, single atom initialization, addressing and readout have been demonstrated in a variety of optical trapping platforms and single-qubit gates have been implemented with exquisite fidelity [1]. Moreover, multi-qubit gates can be implemented with neutral atoms by driving them to highly excited *Rydberg states*. These are states with large principal quantum numbers  $n$ . Consequently, an atom in this state has a diameter which is approximately a factor of  $10^4$  larger than that of a neutral atom in the ground state. However, it is not an ion, as this outer electron is not free, but is still bound, even if very weakly. It is therefore very sensitive to external electromagnetic influences and, consequently, the nearby Rydberg atoms interact very strongly and in a controlled way. This allows for example the realization of conditional quantum logic gates.

In this project, a physical implementation [1] of a two-qubit CZ gate (up to a global gauge choice) between individual neutral atoms trapped in optical tweezer is numerically simulated. More specifically, a CZ gate is an example of a *controlled unitary operator gate*. It has two input qubits, which are designated as the *control* and *target* qubits, respectively. The gate has no effect on the control qubit, but performs a unitary operation on the target qubit conditionally on the state of the control qubit [3]. For instance, for a two-qubit CZ gate the Z operator is applied to the target qubit conditional upon the state of the control qubit, resulting in a matrix representation as in Eq. (1).

$$CZ = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \quad \begin{array}{c} \text{---} \\ \bullet \\ \text{---} \\ \boxed{Z} \\ \text{---} \end{array} \quad (1)$$

First of all, we discuss the physical and theoretical design of the two-qubit CZ gate. Optimal values for the pulse duration  $\tau$ , the detuning  $\Delta$  and the Rabi frequency  $\Omega$  are needed to reproduce exactly its behavior.

Then, we describe in details the numerical simulation of the two-qubit CZ gate both in the case of perfect and imperfect blockade regime. We make use of QuTip, a Python library which provides ready-use functions to handle quantum systems. After that, the CZ gate is generalized to a chain of  $N$ -qubit. In both the two-qubit and chain of  $N$ -qubit system, a Gaussian noise is introduced and noise effects are investigated.

Since for a  $N$ -qubit system the dimension of the total Hamiltonian scales as  $3^N \times 3^N$ , the time evolution of the system can become computationally challenging for larger  $N$ . Hence, methods to perform unitary time evolution in an optimal way are discussed and coded with different implementations. A timing analysis is performed in order to compare different methods and implementations.

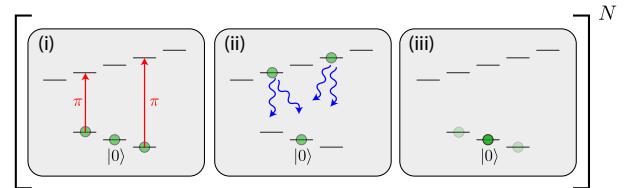
All the implemented code and the results shown in this work are available on GitHub [4].

## II. PHYSICAL IMPLEMENTATION OF SINGLE-QUBIT AND TWO-QUBIT GATES

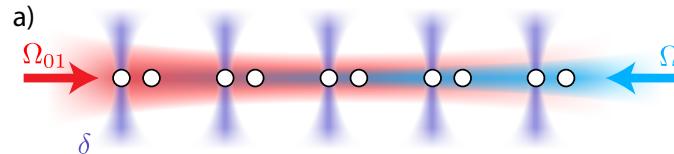
In the real experiment conducted by H. Levine et All. [1], individual neutral atoms are trapped in *optical tweezers* and organized in a one-dimensional array into groups of two. Qubits are encoded in hyperfine ground states of these atoms with  $|0\rangle = |5S_{1/2}, F = 1, m_F = 0\rangle$  and  $|1\rangle = |5S_{1/2}, F = 2, m_F = 0\rangle$ .

**a. Optical pumping into  $|0\rangle$ .** First of all, all qubits are initialized in  $|0\rangle$  through a Raman-assisted optical pumping procedure as illustrated in Fig. 1. Atoms into all  $m_F$  states within the  $|5S_{1/2}, F = 1\rangle$  manifold are coarse pumped and a Raman  $\pi$ -pulse is applied to excite from  $|F = 1, m_F = -1\rangle$  to  $|F = 2, m_F = -1\rangle$  and from  $|F = 1, m_F = +1\rangle$  to  $|F = 2, m_F = +1\rangle$ . Then, the system is coarse pumped back from  $F = 2$  to  $F = 1$ . The effect is that some population from  $|F = 1, m_F = \pm 1\rangle$  is transferred to  $|F = 1, m_F = 0\rangle$ . The cycle is repeated  $N \sim 70$  times to achieve a net population of 99.3% in  $|0\rangle$ . More specifically, the transition between qubit states  $|0\rangle$  and  $|1\rangle$  is driven by a 795 nm Raman laser aligned along the array of atoms. It is  $\sigma^+$  polarized to coherently drive  $\pi$  transitions between  $F = 1$  and  $F = 2$  ground state manifolds with a Rabi frequency of  $\Omega_{01} = 2\pi \times 250$  kHz.

**b. Single-qubit gates.** Single-qubit gates are implemented through a combination of the global 795 nm laser beam which homogeneously and simultaneously drives all qubits in the one-dimensional array and of local 420 nm beams focused onto individual sites, resulting in a light shift  $\delta$  used for individual addressing (Fig. 2). These two coupling are described by  $X(\theta)$  global qubit rotation and  $Z(\theta)$  local rotations. More specifically, the two single-qubit gates physically implemented in the experiment are  $X(\pi/4)$  and  $Z(\pi)$ .



**FIG. 1** Raman-assisted optical pumping scheme (repeated  $N$  times) [1]. (i) All three sublevels of  $F = 1$  are coarse pumped and a Raman  $\pi$ -pulse is applied to excite from  $|F = 1, m_F = -1\rangle$  to  $|F = 2, m_F = -1\rangle$  and from  $|F = 1, m_F = +1\rangle$  to  $|F = 2, m_F = +1\rangle$ . (ii) The system is coarse pumped back from  $F = 2$  to  $F = 1$ . (iii) Some population from  $|F = 1, m_F = \pm 1\rangle$  is transferred to  $|F = 1, m_F = 0\rangle$ .



**FIG. 2** One-dimensional array of neutral atoms [1]. Atoms are arranged in pairs and are globally driven with a 795 nm Raman laser (in red) which couples the qubit states  $|0\rangle$  and  $|1\rangle$ . Local 420 nm (purple) beams are focused onto individual atoms. Atoms are globally excited by a bi-chromatic, 420 nm and 1013 nm, Rydberg laser (blue) from the  $|1\rangle$  qubit state to  $|r\rangle$ .

For a two qubits system, both initialized in 0, all the four computational basis states are experimentally initialized using global  $X(\pi/2)$  pulses (two sequential  $X(\pi/4)$  gates) and local  $Z(\pi)$  gates on the left atom only (top qubit in each circuit):

- the  $|00\rangle$  requires no pulses to prepare;
- the  $|01\rangle$  state is prepared as follows:

$$|0\rangle \xrightarrow{\boxed{X(\pi/2)}} \boxed{Z(\pi)} \xrightarrow{\boxed{X(\pi/2)}} |0\rangle$$

$$|0\rangle \xrightarrow{\boxed{X(\pi/2)}} \xrightarrow{\boxed{X(\pi/2)}} |1\rangle$$

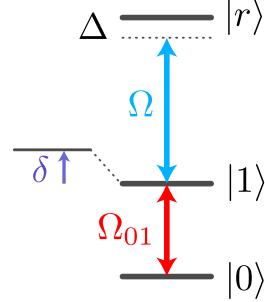
- the  $|10\rangle$  state is prepared as follows:

$$|0\rangle \xrightarrow{\boxed{X(\pi/2)}} \boxed{Z(\pi)} \xrightarrow{\boxed{X(3\pi/2)}} |1\rangle$$

$$|0\rangle \xrightarrow{\boxed{X(\pi/2)}} \xrightarrow{\boxed{X(3\pi/2)}} |0\rangle$$

- the  $|11\rangle$  state requires only a global  $X(\pi)$  gate.

**c. Two-qubit gates.** Two-qubit gates are physically implemented by exciting atoms from the qubit state  $|1\rangle$  to the Rydberg state  $|r\rangle = |70S_{1/2}, m_J = -1/2\rangle$  through a two-photon process with effective Rabi frequency  $\Omega \approx 2\pi \times 3.5$  MHz. The coupling from  $|1\rangle$  to  $|r\rangle$  is obtained through a bi-chromatic laser system. In particular, the laser at 420 nm is polarized to drive a  $\sigma^-$  transition through the intermediate state  $|6P_{3/2}\rangle$ , while the one at 1013 nm to drive a  $\sigma^+$  transition. The relevant atomic levels for realizing two-qubit gates are summarized in Fig. 3.



**FIG. 3** Relevant atomic levels [1]. The qubit states  $|0\rangle = |5S_{1/2}, F = 1, m_F = 0\rangle$  and  $|1\rangle = |5S_{1/2}, F = 2, m_F = 0\rangle$  are coupled with Rabi frequency  $\Omega_{01}$ .  $\delta$  is the light shift used for individual addressing resulting from local 420 nm beams. The qubit state  $|1\rangle$  is coupled to the Rydberg state  $|r\rangle = |70S_{1/2}, m_J = -1/2\rangle$  with detuning  $\Delta$  and effective Rydberg Rabi frequency  $\Omega$ .

A new protocol for implementing a two-qubit **controlled-phase (CZ) gate** consists of just two global laser pulses (with the laser phase of the second pulse shifted by  $\xi$ ) of the same length  $\tau$  and detuning  $\Delta$  which couple  $|1\rangle$  to  $|r\rangle$  and drive nearby atoms within the Rydberg blockade regime. Indeed, within a given cluster of atoms the strength of the Rydberg interaction is strong, i.e.  $2\pi \times 24$  MHz  $\gg \Omega$ , since atoms are trapped sufficiently close to each other. Therefore, neighboring atoms cannot be simultaneously excited to the Rydberg state according to the *Rydberg blockade*. The desired CZ gate (up to global rotation of qubits) maps the computational basis states as follows:

$$\begin{aligned} |00\rangle &\rightarrow |00\rangle \\ |01\rangle &\rightarrow |01\rangle e^{i\phi} \\ |10\rangle &\rightarrow |10\rangle e^{i\phi} \\ |11\rangle &\rightarrow |11\rangle e^{i(2\phi-\pi)} \end{aligned} \quad (2)$$

The behavior of the four computational basis states is:

- the  $|00\rangle$  state does not evolve since it is uncoupled by the laser field;
- the dynamics of  $|01\rangle$  (and  $|10\rangle$ ) are given by the coupling of the single atom on the  $|1\rangle \leftrightarrow |r\rangle$  transition forming a two-level system with Rabi frequency  $\Omega$  and detuning  $\Delta$  (Fig. 4a);
- the  $|11\rangle$  state evolves within the Rydberg blockade regime as a two-level system due to the collective coupling from  $|11\rangle \leftrightarrow |W\rangle = \frac{1}{\sqrt{2}}(|1r\rangle + |r1\rangle)$ , with enhanced Rabi frequency  $\sqrt{2}\Omega$  and the same detuning  $\Delta$  (Fig. 4b).



**FIG. 4** The dynamics of  $|01\rangle$  and  $|11\rangle$  states can be described by two-level systems with the same detuning but different effective Rabi frequencies. For a fixed detuning, the pulse length  $\tau$  is selected such that the first laser pulse drives an incomplete Rabi oscillation on the  $|01\rangle$  system, while the  $|11\rangle$  system undergoes a complete detuned Rabi cycle. The phase of the second pulse  $\xi$  is chosen such that for the  $|01\rangle$  system a second pulse of length  $\tau$  completes the oscillation and returns to the state  $|01\rangle$ . For the  $|11\rangle$  system the second pulse drives a second complete cycle.

For a fixed detuning  $\Delta$ , the pulse length  $\tau$  is chosen such that:

1. the *first laser pulse* drives an incomplete oscillation on the  $|01\rangle$  system, while the  $|11\rangle$  system completes a full cycle of a detuned Rabi oscillation;
2. the driving field of the *second laser pulse* is rotated around the  $z$ -axis by an angle  $\xi$  such that a second pulse of length  $\tau$  completes the oscillation for a  $|01\rangle$  system returning to  $|01\rangle$ . It also drives a second complete oscillation on the  $|11\rangle$  configuration.

After the second pulse, both  $|01\rangle$  and  $|11\rangle$  systems return to their initial positions on the Bloch sphere but with an accumulated dynamical phases  $\phi_{01}$  and  $\phi_{11}$  (Fig. 4, 5). The acquired dynamical phase depends on the solid angle enclosed by the  $\Delta$ -dependent trajectories on the Bloch sphere. It varies from  $2\pi$  to 0 with increasing  $\Delta$ . Specifically, with a choice of the laser detuning as  $\Delta \approx 0.377 \Omega$ , the dynamical phase  $\phi_{11} = 2\phi_{01} - \pi$ , which realizes the CZ gate described in Eq. (2).



**FIG. 5** The dynamical phases  $\phi_{01}$  and  $\phi_{11}$  are determined by the solid angle enclosed by the trajectories on the Bloch sphere (shaded area). They vary from  $2\pi$  to 0 as a function of  $\Delta$  corresponding to increasingly shallow trajectories. The CZ gate mapped in (2) is realized by choosing  $\Delta \approx 0.377 \Omega$ .

### III. THEORETICAL DESIGN OF TWO-QUBIT CZ GATE

The dynamics of a two-qubit CZ gate, implemented with the protocol described in Sec. II, is governed by the following Hamiltonian:

$$H = \sum_{i=1}^2 \left[ \frac{1}{2} (\Omega |1\rangle_i \langle r| + \Omega^* |r\rangle_i \langle 1|) - \Delta |r\rangle_i \langle r| \right] + V |r\rangle_1 \langle r| \otimes |r\rangle_2 \langle r| \quad (3)$$

where  $\Delta$  is the detuning of the excitation laser from the transition frequency between states  $|1\rangle$  and  $|r\rangle$ , and  $\Omega$  is the Rabi frequency.  $V$  is the interaction strength between two atoms in Rydberg states.

**a. Perfect blockade.** If the interaction between two atoms in Rydberg states is strong, i.e.  $V \gg |\Omega|, |\Delta|$ , the system is in the Rydberg blockade regime. The dynamics of the system simplifies as:

- the state  $|00\rangle$  does not evolve;
- if one of the two atoms is in  $|0\rangle$ , only the system in  $|1\rangle$  evolves. The dynamics can be described with a two level system with states  $|a_1\rangle \equiv |1\rangle$  and  $|b_1\rangle \equiv |r\rangle$  and Hamiltonian:

$$H_1 = \frac{1}{2} (\Omega |a_1\rangle \langle b_1| + \Omega^* |b_1\rangle \langle a_1|) - \Delta |b_1\rangle \langle b_1| \quad (4)$$

- if both atoms are initially in  $|1\rangle$ , the dynamics can be described with a two level system with states  $|a_2\rangle \equiv |11\rangle$  and  $|b_2\rangle \equiv \frac{1}{\sqrt{2}}(|r, 1\rangle + |1, r\rangle)$  and Hamiltonian:

$$H_2 = \frac{\sqrt{2}}{2} (\Omega |a_2\rangle \langle b_2| + \Omega^* |b_2\rangle \langle a_2|) - \Delta |b_2\rangle \langle b_2| \quad (5)$$

As explained in details in Sec. II, the CZ gate is implemented with two identical global pulses of the Rydberg laser field with equal duration  $\tau$  and detuning  $\Delta$  and with a phase jump by  $\xi$  in between. Each pulse is mathematically represented by an unitary evolution of the state:

$$U = \exp(-iH\tau) \quad (6)$$

Moreover, the change of the laser phase between the two pulses is represented by a rotation by an angle  $\xi$  around the  $z$ -axis,  $\Omega \rightarrow \Omega e^{i\xi}$ . Finally, the operator  $\mathcal{U}$  is defined as the application of both laser pulses.

The computational basis states evolve as follows under the action of  $\mathcal{U}$ :

- the state  $|00\rangle$  does not evolve:

$$\mathcal{U} |00\rangle = |00\rangle \quad (7)$$

- the state  $|11\rangle$  evolves as:

$$\mathcal{U} |11\rangle = e^{i\phi_2} |11\rangle \quad (8)$$

where the accumulated dynamical phase is  $\phi_2 = 2\pi \times 2\Delta/\sqrt{\Delta^2 + 2\Omega^2}$ . Indeed, if we select the pulse duration  $\tau$  such that:

$$\tau = \frac{2\pi}{\sqrt{\Delta^2 + 2\Omega^2}} \quad (9)$$

the first laser pulse drive a complete detuned Rabi cycle for the  $|11\rangle$  state, namely  $U |11\rangle = e^{i\phi_2/2} |11\rangle$ . The second pulse also leads to a complete Rabi cycle about a different axis, but results in the same accumulated phase. In total, we find that the evolution of the system under the application of both laser pulses is as Eq. (8).

- the states  $|01\rangle$  or  $|10\rangle$  evolve as:

$$\begin{aligned}\mathcal{U}|01\rangle &= e^{-i\phi_1}|01\rangle \\ \mathcal{U}|10\rangle &= e^{-i\phi_1}|10\rangle\end{aligned}\tag{10}$$

Indeed, after the first pulse, due to the mismatch between the effective Rabi frequencies in  $H_1$  and  $H_2$ , the state  $|01\rangle$  (or  $|10\rangle$ ) does not return to itself, but a superposition state is created:

$$\begin{aligned}U|01\rangle &= \cos(\alpha)|01\rangle + \sin(\beta)e^{i\gamma}|0r\rangle \\ U|10\rangle &= \cos(\alpha)|10\rangle + \sin(\beta)e^{i\gamma}|r0\rangle\end{aligned}\tag{11}$$

With a proper choice of the phase jump  $\xi$  between the two pulses, the system returns to the state  $|01\rangle$  (or  $|10\rangle$ ) after the second pulse. This proper choice is:

$$e^{-i\xi} = \frac{-\sqrt{(\Delta/\Omega)^2 + 1} \cos\left(\frac{1}{2}\Omega\tau\sqrt{(\Delta/\Omega)^2 + 1}\right) + i\Omega\tau \sin\left(\frac{1}{2}\Omega\tau\sqrt{(\Delta/\Omega)^2 + 1}\right)}{\sqrt{(\Delta/\Omega)^2 + 1} \cos\left(\frac{1}{2}\Omega\tau\sqrt{(\Delta/\Omega)^2 + 1}\right) + i\Omega\tau \sin\left(\frac{1}{2}\Omega\tau\sqrt{(\Delta/\Omega)^2 + 1}\right)}\tag{12}$$

Both the acquired dynamical phases  $\phi_1$  and  $\phi_2$  are solely determined by the dimensionless quantity  $\Delta/\Omega$ . Finally, a choice for  $\Delta/\Omega$  can be found in order to have  $\phi_2 = 2\phi_1 + \pi$ . With this choice we reproduce exactly the behavior of controlled-phase gate in Eq. (2). The numerical values of the optimal parameters which describes this dynamics are:

$$\Omega\tau = 4.29268\tag{13a}$$

$$\xi = 3.90242\tag{13b}$$

$$\Delta/\Omega = 0.377371\tag{13c}$$

**b. Accounting for imperfect blockade.** If finite blockade interactions are accounted for, the finite value of  $V$  only affects the dynamics if the system is initially in the state  $|11\rangle$ . Indeed, the dynamics of a system initially in  $|11\rangle$  can be described with a three level system with states  $|a_2\rangle \equiv |11\rangle$ ,  $|b_2\rangle \equiv |1r\rangle + |r1\rangle$  and  $|c_2\rangle = |rr\rangle$  and Hamiltonian:

$$H_2 = \frac{\sqrt{2}}{2}(\Omega|a_2\rangle\langle b_2| + \Omega|b_2\rangle\langle c_2| + \Omega^*|c_2\rangle\langle b_2| + \Omega^*|b_2\rangle\langle a_2|) - \Delta|b_2\rangle\langle b_2| + (V - 2\Delta)|c_2\rangle\langle c_2|\tag{14}$$

For small  $V > 0$  and a given  $\Delta$ , a value for  $\Omega$  and  $\tau$  can always be chosen such that a system initialized in  $|11\rangle$  returns to itself after just the first pulse, as in the perfect blockade assumption. Thus in the finite blockade regime the complete Rabi oscillation are simply replaced by a slightly more complicated path, but which is still closed in the Bloch sphere.

For  $V \gg |\Delta|, |\Omega|$ , the results of the perfect blockade regime are obtained. Indeed, the effect for finite blockade simply reduces to the two-level system,  $\{|a_2\rangle, |b_2\rangle\}$ , where the detuning  $\Delta$  is renormalized by  $\Omega^2/(2V)$ .

#### IV. CODE IMPLEMENTATION OF TWO-QUBIT CZ GATE

A numerical simulation of the two-qubit CZ gate, described in details in Sec. III, is performed. First of all, a two-qubit system is implemented in order to investigate the dynamics of a cluster of two atoms under the action of the controlled-phase gate. Then, we implement a chain of  $N$ -qubit where the CZ gate is applied between consequent qubits under the assumption that there is interaction only within an atom pair. This is useful for a further analysis to study how a Gaussian noise for the optimal parameters in Eq. (13) propagates through the chain.

##### 1. Two-qubit system

To simulate the dynamics of a two-qubit system under the application of the two-qubit CZ gate, the Python library QuTip is used. This library provides ready-use functions to handle quantum systems and also provide optimized solvers for time evolution of Schrödinger's equations. The system is composed by two atoms each one can be in states  $|0\rangle$ ,  $|1\rangle$  or  $|r\rangle$ . Thus system state vectors have dimension  $d = 3^2$ .

**a. Perfect blockade.** Under the assumption of a perfect blockade regime, the dynamics of the system decouples into a few simple sectors described by the Hamiltonians in Eqs. (4) and (5). The total Hamiltonian of the system,  $H_{CZ}$ , has dimension  $3^2 \times 3^2$  and it is implemented in the Python function `hamiltonian(Omega,Delta)`. The CZ gate is implemented in function `CZ_gate(psi,Omega,Delta,tau)`. In more details, the time between 0 and  $\tau$  is discretized and each pulse is applied to the system as a unitary evolution (Eq. (6)). In particular, the second pulse is rotated by  $\Omega \rightarrow \Omega^{i\xi}$ . The unitary evolution of the system state is calculated using the QuTiP function `qutip.mesolve`. It evolves the state vector in the list of times, using an ordinary differential equation solver. Using this function, also expectation values for a set of operator can be evaluated in time.

```
# Hamiltonian definition
def hamiltonian(Omega,Delta):

    H0 = 0 * tensor(psi00.dag(),psi00)

    H01 = 1/2 * ( Omega*tensor(psi01.dag(),psi0r) + np.conj(Omega)*tensor(psi0r.dag(),psi01) )
                - Delta*tensor(psi0r.dag(),psi0r)

    H10 = ...
    H2 = ...

    H = H0 + H01 + H10 + H2

    return H

# CZ gate implementation
def CZ_gate(psi,Omega,Delta,tau):

    # Times discretization
    times = np.linspace(0.0, tau, 200)

    # Apply first pulse
    H = hamiltonian(Omega,Delta)
    result = mesolve(H, psi, times,[],[])
    psi = result.states[-1]

    # Apply second pulse rotated by Omega -> Omega exp(i xi)
    H = hamiltonian(Omega * exp_xi(Delta,Omega,tau), Delta)
    result = mesolve(H, psi, times,[],[])
    psi = result.states[-1]

    return result
```

**b. Imperfect blockade.** If finite blockade interactions are taken into account, only the dynamics of the system initially in  $|11\rangle$  is affected and the Hamiltonian for this sector becomes as in Eq. (14). The total Hamiltonian is implemented in the function `hamiltonian_imperfect(Omega,Delta,V)`. The dynamics of the system is simulated with the QuTip function `qutip.mesolve` as for the perfect Rydberg blockade case.

## 2. Chain of N-qubit

We implement a numerical simulation of a chain of  $N$ -qubit where the CZ gate is applied between consequent qubits, assuming perfect blockade regime. Physically, it is like considering a one-dimensional array of atoms where, each time the two-qubit CZ gate acts on a pair of atoms, the interaction with the other atoms is neglected. In order to obtain the behavior of the map in Eq. (2) between the first and last qubits, they can be initialized either in state  $|0\rangle$  or  $|1\rangle$ , while all the other qubits in the chain are initialized in  $|0\rangle$ .

The state vector of the  $N$ -qubit system is computed as the tensor product of the subsystems as:

$$|\psi\rangle = |q_1\rangle \otimes \cdots \otimes |q_N\rangle \quad (15)$$

The function `chain.init(N,state_first,state_last)` computes the initial state of the chain, given the initial state of the first and the last qubits. Then, in the function `chain.CZ_gate(psi,Omega,Delta,tau,method,niter)` the CZ gate in a chain of  $N$ -qubit is implemented. Starting from the first qubit, we apply the two-qubit CZ gate iteratively. In the  $i$ -th iteration, the total Hamiltonian of the chain,  $H_{chain} \in 3^N \times 3^N$ , let evolve the  $i$ -th and the  $(i+1)$ -th qubits without affecting the remaining  $N-2$ . It is computed as:

$$H_{chain,i}^{(N)} = \underbrace{\mathbb{I}_3 \otimes \cdots \otimes \mathbb{I}_3}_{i-1} \otimes H_{CZ} \otimes \underbrace{\mathbb{I}_3 \otimes \cdots \otimes \mathbb{I}_3}_{N-i-1}. \quad (16)$$

```
# Chain state initialization
def chain_init(N,state_first,state_last):

    psi = basis(3,state_first)
    for i in range(N-2):
        psi = tensor(psi,basis(3,0))
    psi = tensor(psi,basis(3,state_last))

    return psi

# Chain CZ-gate function
def chain_CZ_gate(psi,Omega,Delta,tau,method,niter):

    N = len(dims(psi)[0]) # Number of qubits in the chain
    I = qeye(3) # Define identity matrix
    Hp1 = Qobj( hamiltonian(Omega,Delta) ) # First pulse hamiltonian
    Hp2 = Qobj( hamiltonian(Omega * exp_xi(Delta,Omega,tau), Delta) ) # Second pulse hamiltonian
    times = np.linspace(0.0, tau, niter) # Discretize time

    for i in range(N-1):

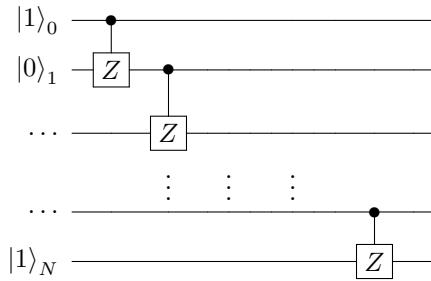
        mat_list1 = [I]*(N-1)
        mat_list2 = [I]*(N-1)
        mat_list1[i] = Hp1
        mat_list2[i] = Hp2

        H1 = mat_list1[0]
        H2 = mat_list2[0]

        for i in range(N-2):
            H1 = tensor(H1,mat_list1[i+1])
            H2 = tensor(H2,mat_list2[i+1])

        result = mesolve(H1, psi, times,[],[])
        psi = result.states[-1]
        result = mesolve(H2, psi, times,[],[])
        psi = result.states[-1]

    return psi
```



**FIG. 6** Circuit scheme of CZ gate in a chain of  $N$ -qubit.

## V. NUMERICAL METHODS FOR TIME-DEPENDENT SCHRÖDINGER EQUATION SOLVERS

For the  $N$ -qubit system discussed in Sec. IV.2, as longer chains are considered, the dimension of the total Hamiltonian of the system scales as  $3^N \times 3^N$ . Given the Hamiltonian  $H$  of the system, the time-dependent Schrödinger equation can be solved as:

$$|\psi(t)\rangle = U(t) |\psi(0)\rangle, \quad U(t) = e^{-iHt/\hbar} \quad (17)$$

The time evolution of the system can become computationally challenging for large  $N$ . Therefore, we consider several time-dependent Schrödinger solvers, along with some optimizations, in order to test their performances. We implement these algorithms using different coding environment and libraries implementation both on CPU and GPU.

**a. Numpy implementation** First of all, we implement a script analogous to the one described in Sec. IV.2, but based on NumPy and SciPy Python libraries. Numpy provides support for large, multi-dimensional arrays, along with a collection of high-level mathematical functions to operate on these arrays. In particular, NumPy is coded in well-optimized C code, so it allows to exploit the flexibility of Python and the speed of compiled code. It guarantees generally faster execution time compared to Python built in types. SciPy integrates and expand Numpy tools providing additional functions to operate with matrices, as well as sparse array libraries. Eventually, NumPy, along with SciPy, supports a wide range of hardware and computing platforms and plays well with parallel computing.

**b. Tensorflow implementation** Then, we implement again the code using TensorFlow, a Python library for machine learning, which is optimized to deal with large matrices and in particular can be easily executed on GPU thanks to CUDA.

**c. Fortran implementation** Finally, the same code is implemented using Fortran programming language. Indeed, it is most of the most powerful language for scientific computation with LAPACK routines for linear algebra. The design of Fortran allows the compiler to perform stronger optimizations.

### 1. Spectral method

We compute the unitary time evolution of the system as in Eq. (17). To do this, we first diagonalize the system Hamiltonian with the decomposition  $H = PDP^{-1}$ , where  $D$  is the diagonal matrix and  $P$  is the eigenvectors matrix. Its exponential can be computed as:

$$e^{-iHt} = Pe^{-iDt}P^{-1} \quad (18)$$

This method requires the computation of eigenvalues and eigenvectors of a matrix and the inversion of a matrix.

In Numpy, we implement the function `spectral_evol(H, psi, tau)` in which the eigenvalues decomposition is achieved through the function `numpy.linalg.eigh`, optimized for complex Hermitian matrices, which is implemented using `_syevd` and `_heevd` LAPACK routines for computing eigenvalues and eigenvectors, respectively. The inversion of the matrix is achieved through the function `numpy.linalg.inv` which again relies on LAPACK routines.

```
# Time evolution with spectral method in Numpy
def spectral_evol(H, psi, tau):

    eiglist = np.linalg.eigh(H)
    eigenvalues = eiglist[0]
    eigenvectors = eiglist[1]
    eigenvectors_inv = np.linalg.inv(eigenvectors)

    exp_diag = np.diag(np.exp(-1j*eigenvalues*tau))

    exp_H = np.dot(eigenvectors, np.dot(exp_diag, eigenvectors_inv))

    return np.dot(exp_H, psi)
```

In TensorFlow, the unitary time evolution operator is computed by means of the function `tensorflow.linalg.expm` to compute the exponential matrix.

Finally, in Fortran, implementation eigenvalues decomposition is achieved through `_heev` and the matrix inversion is obtained with `_getrf` and `_getri` LAPACK routines based on LU decomposition.

## 2. Crank-Nicolson method

The time evolution of the system is also performed using Crank-Nicolson algorithm. Given a discrete time step  $\Delta t$ , the unitary time evolution operator can be written as:

$$e^{-iH\Delta t} = e^{-\frac{iH\Delta t}{2}} e^{-\frac{iH\Delta t}{2}} = \left( e^{\frac{iH\Delta t}{2}} \right)^{-1} e^{-\frac{iH\Delta t}{2}} \simeq \left( 1 + \frac{iH\Delta t}{2} \right)^{-1} \left( 1 - \frac{iH\Delta t}{2} \right) + O(\Delta t)^3 \quad (19)$$

Thus the time evolved wave function can be computed as

$$\psi(x, t + \Delta t) = \left( 1 + \frac{iH\Delta t}{2} \right)^{-1} \left( 1 - \frac{iH\Delta t}{2} \right) \psi(x, t). \quad (20)$$

This is equivalent to solve the linear system:

$$\left( 1 + \frac{iH\Delta t}{2} \right) \psi(x, t + \Delta t) = \left( 1 - \frac{iH\Delta t}{2} \right) \psi(x, t). \quad (21)$$

By iterating the procedure  $n_{iter}$  times, we obtain the time evolution for time  $T = \Delta t \times n_{iter}$ .

In Numpy, the Crank-Nicolson method is implemented in the function `crank_nicolson(H, psi, tau, n_iter)`. Given the Hamiltonian, it computes the time evolution of the initial state  $\psi$  for a time  $\tau$  using  $n_{iter}$  iterations. In this algorithm we need to solve a linear system of equations. To do this we exploit the function `numpy.linalg.solve` which uses LAPACK routine `gesv`.

```
# Crank-Nicolson solver in Numpy
def crank_nicolson(H, psi, tau, n_iter):

    time = np.linspace(0, tau, n_iter)
    dt = time[1] - time[0]

    A = np.eye(H.shape[0], dtype=data_type) + 1j * H * dt/2
    B = np.eye(H.shape[0], dtype=data_type) - 1j * H * dt/2

    b = np.dot(B, psi)

    for i in time:

        psi = np.linalg.solve(A, b)
        b = np.dot(B, psi)

    return psi
```

For TensorFlow implementation we write an analogous function using `tensorflow.linalg` module.

## 3. Crank-Nicolson method with LU decomposition

In order to optimize Crank-Nicolson algorithm we consider the LU decomposition method for solving linear systems. Indeed, an invertible matrix  $A$  can be decomposed into two factors  $LU$ , where  $L$  is a lower and a  $U$  an upper triangular matrix. Thus the linear system  $Ax = y$  can be recasted as:

$$\begin{cases} Lz = y \\ Ux = z \end{cases} \quad (22)$$

In our case the matrix  $A$  is fixed and the system has to be solved many times for different  $x$  and  $y$ , so this procedure is convenient because we can execute the LU decomposition just once. Then, to solve the systems only two back substitutions are required.

In Numpy, we define the function `crank_nicolson_LU(H, psi, tau, n_iter)`. Again, given the Hamiltonian, it computes the time evolution of the initial state  $\psi$  for a time  $\tau$  using  $n_{iter}$  iterations. We compute the LU decomposition through `scipy.linalg.lu` function and we solve the systems we exploiting the `scipy.linalg.solve_triangular` function optimized for triangular matrices.

```
# Crank-Nicolson LU optimized solver in Numpy
def crank_nicolson_LU(H, psi, tau, n_iter):

    time = np.linspace(0, tau, n_iter)
    dt = time[1] - time[0]

    A = np.eye(H.shape[0], dtype=data_type) + 1j * H * dt/2
    B = np.eye(H.shape[0], dtype=data_type) - 1j * H * dt/2

    b = np.dot(B, psi)

    L, U = scipy.linalg.lu(A, permute_l = True)

    for i in time:

        y = scipy.linalg.solve_triangular(L,b,lower=True)
        psi = scipy.linalg.solve_triangular(U,y,lower=False)
        b = np.dot(B,psi)

    return psi
```

As before, for TensorFlow implementation we write an analogous function using **tensorflow.linalg** module.

Finally, we can observe that the total Hamiltonian of the  $N$ -qubits system is a *sparse matrix*. Therefore, we optimize the Crank-Nicolson algorithm with LU decomposition for the case of sparse matrices. We implement the function **crank\_nicolson\_LU\_sparse** ( $H$ ,  $\psi$ ,  $\tau$ ,  $n_{\text{iter}}$ ) which is analogous to **crank\_nicolson\_LU**. Now, the matrices are transformed into **csc\_matrix** sparse matrices and optimized functions are used for LU decomposition and triangular systems solving. Instead, no sparse matrix optimized solvers are provided with TensorFlow.

## VI. RESULTS

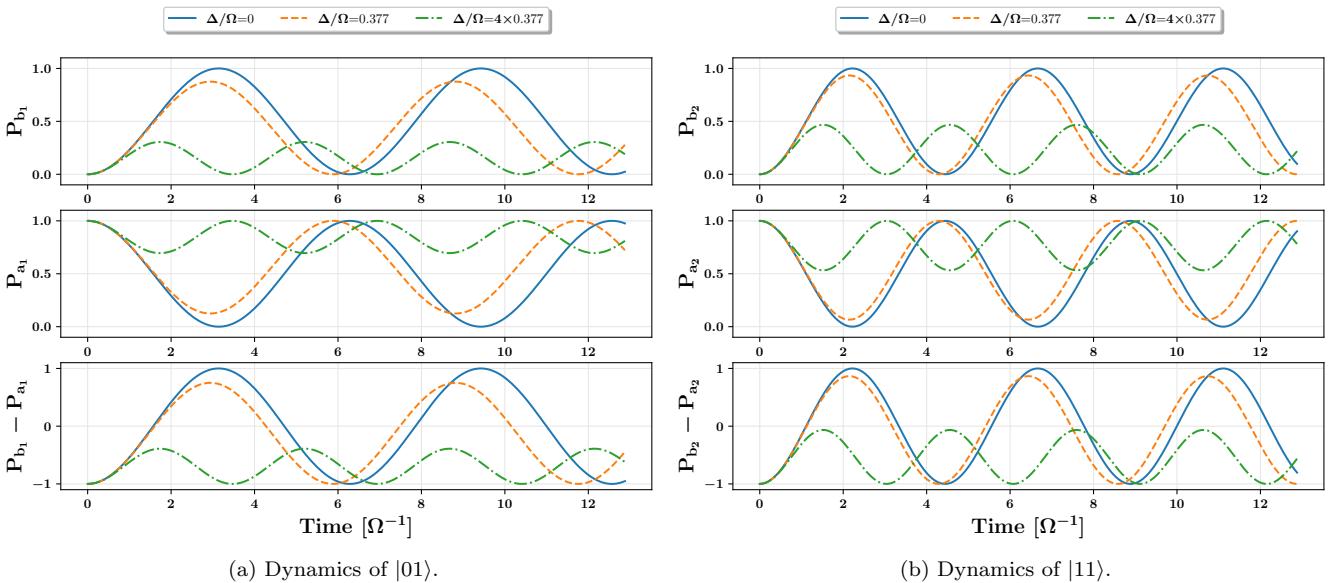
The behavior of the CZ gate, described theoretically in Sec. III, is numerically simulated as explained in Sec. IV. Firstly, we analyze the CZ gate acting on a two-qubit system in both the perfect and imperfect blockade regime. The correctness of the simulation is verified by fixing the optimal parameters (Eq. (13)) and comparing the results with the experimental one discussed in Sec. II. Then, we vary the optimal parameters in order to investigate the behavior of the CZ gate in response to this variation. After that, the measurement process is simulated by introducing a Gaussian noise on the parameters in the case of perfect and imperfect blockade regime. Then, we consider CZ gate in the case of a chain of  $N$ -qubit. Again, a Gaussian noise is introduced on the optimal parameters to investigate noise effects with an increasing number of qubits. Finally, a timing analysis of the several algorithms for time evolution and implementations, discussed in Sec. V, is performed.

### 1. CZ gate in a two-qubit system

Let us consider a system of two-qubit with the CZ gate acting between them. In more details, the system is composed by two atoms each one can be in states  $|0\rangle$ ,  $|1\rangle$  and  $|r\rangle$ . The dynamics is analyzed both in the case of perfect blockade regime or finite blockade interactions.

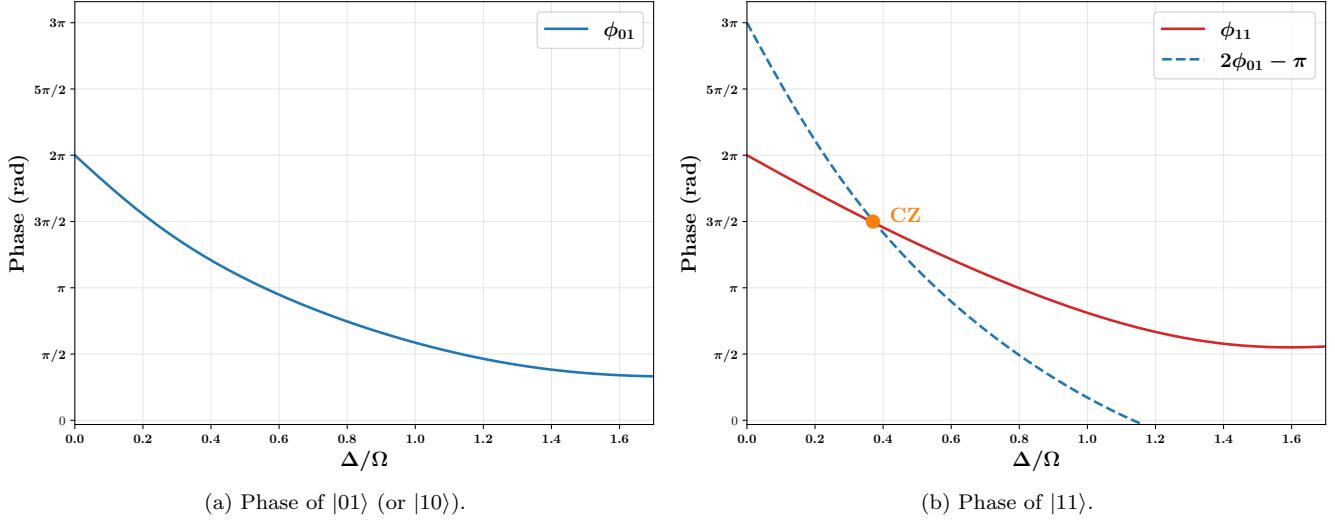
#### A. Correct behaviour check and optimal parameters variation

**a. Perfect blockade.** As explained in details in Sec. II and III, under the assumption of a perfect Rydberg blockade, the dynamics of the states  $|01\rangle$  (or  $|10\rangle$ ) and  $|11\rangle$  can be simplified as the one of a two-level system with same detuning  $\Delta$  and different Rabi frequencies. In particular, the Rabi frequency of  $|01\rangle$  (or  $|10\rangle$ ) is  $\Omega$ , while the one of  $|11\rangle$  has an enhanced Rabi frequency  $\sqrt{2}\Omega$ , as one can see by comparing the Eqs. (4) and (5). More specifically, the dynamics, under the application of the first pulse of duration (13a), is simulated in Fig. 7 for different  $\Delta/\Omega$  ratios. If the system is initially in the state  $|01\rangle$  (or  $|10\rangle$ ), only the atoms that it is in  $|1\rangle$  evolves and oscillates between the states  $|a_1\rangle \equiv |1\rangle$  and  $|b_1\rangle \equiv |r\rangle$  (Fig. 7a). One can note that for null detuning (blue line) the oscillation of the quantum system is most efficient. In this case, the population inversion oscillates between -1 and +1. For the ratio chosen as optimal parameters in Eq. (13c) (orange line), the amplitude of the oscillation is slightly reduced. Then, for an extreme value of  $\Delta/\Omega$ , we note that the oscillations have an high frequency. If the system is initially in the state  $|11\rangle$ , then the dynamics is again equivalent to that of an effective two-level system formed by the states  $|a_2\rangle \equiv |11\rangle$  and  $|b_2\rangle \equiv \frac{1}{\sqrt{2}}(|r, 1\rangle + |1, r\rangle)$ .



**FIG. 7** Time-dependence of the populations and the inversion for the two-level systems in Eqs. (4) and (5) under Rydberg perfect blockade assumption for different  $\Delta/\Omega$  ratios. The duration of the pulse is fixed by the product  $\Omega\tau = 4.293$  (Eq. (13a)).

Moreover, the duration of the pulse  $\tau$  is chosen such that the state  $|11\rangle$  evolve as Eq. (8) after the two pulses, i.e. it return to itself and acquire a dynamical phase  $\phi_{11}$ . Then, the proper choice of the phase jump between the two pulses  $\xi$  as in Eq. (12) guarantees that the system initially in  $|01\rangle$  return to itself after the second pulse (the same for  $|10\rangle$ ) with a dynamical phase  $\phi_{01}$ . Finally, there is a proper choice for  $\Delta/\Omega$  such that  $\phi_{11} = 2\phi_{01} - \pi$ , in order to realize a controlled-phase gate up to a global gauge rotation. The phase evolution as a function of the  $\Delta/\Omega$  ratio is plotted in Fig. 8. This behavior resemble exactly what we have already shown in Fig. 5. Thus, the CZ gate behaves correctly as expected from the theory discussed in Sec. III by choosing the optimal parameters in (13).



**FIG. 8** Dynamical phase  $\phi_{01}$  (or  $\phi_{10}$ ) and  $\phi_{11}$  acquired respectively by the states  $|01\rangle$  (or  $|10\rangle$ ) and  $|11\rangle$  after the application of the two pulses. The intersection between  $\phi_{11}$  and  $2\phi_{01} - \pi$  shown in 8b realizes the CZ gate.

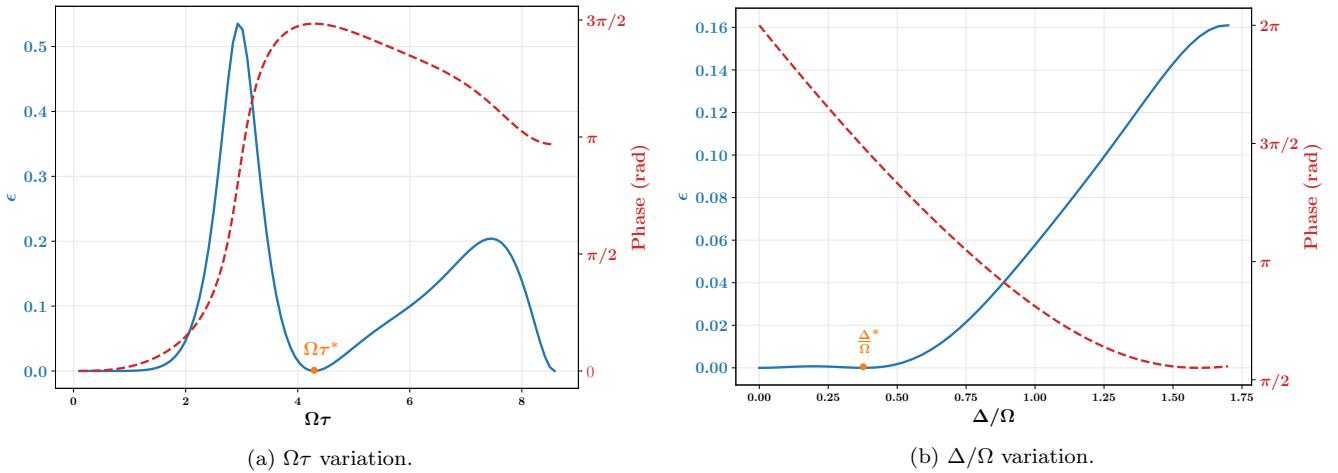
After verifying the correctness of the CZ gate implementation, we vary the optimal parameters in order to investigate the behavior of the controlled-phase gate in different conditions. As already said in details, the key point for a controlled-phase gate implementation is that, after the application of the two pulses, a system initially in  $|01\rangle$  (or  $|10\rangle$ ) and  $|11\rangle$  return to itself with an acquired dynamical phase. To estimate the error on the state of the system, we perform fidelity measurements. More specifically, we will refer as *error* to the following quantity:

$$\epsilon \equiv 1 - F(\rho_{init}, \rho_{fin}) \quad (23)$$

where  $F$  is the fidelity between the initial  $\rho_{init}$  and final state  $\rho_{fin}$ . Moreover, we compute the phase of the final state to monitor its value.

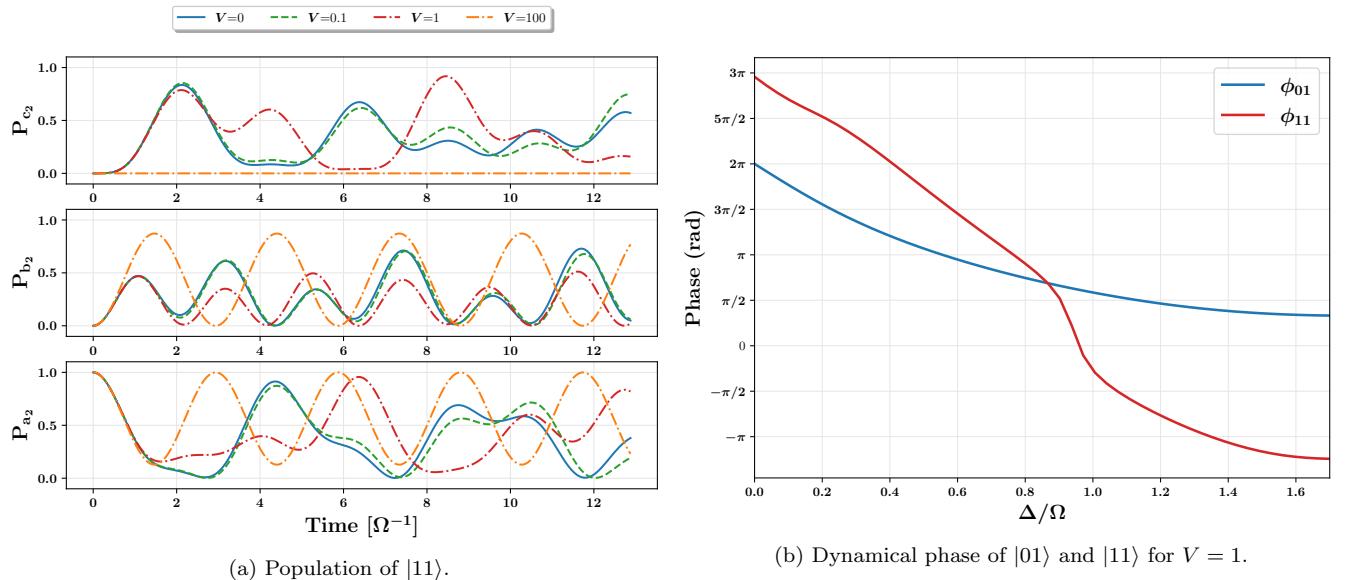
First of all, the quantity  $\Omega\tau$  is changed and the error  $\epsilon$  is measured. If the system is initially in  $|01\rangle$  (or  $|10\rangle$ ) the error remains approximately null as a function of  $\Omega\tau$ . Indeed, in this case, the state return to itself with a very low error regardless  $\Omega\tau$ . However, the dynamical phase  $\phi_{01}$  acquired is still influenced by this choice. Instead, the result for a pair of atom initially in  $|11\rangle$  is shown in Fig. 9a. One can note a peak of  $\epsilon = 0.5$  between  $2 < \Omega\tau < 4$ . Another peak is present between  $6 < \Omega\tau < 8$ . Hence, the optimal parameter choice is in between the two peaks. Thus, we can conclude that an optimal choice for  $\Omega\tau$  is required in order to implement a CZ gate correctly. For the sake of completeness, in the same plot also the evolution of the dynamical phase  $\phi_{11}$  is shown.

Then, we vary the quantity  $\Delta/\Omega$  in order to investigate the response of the system. In Fig. 9b, the error  $\epsilon$ , for a system initially in  $|11\rangle$ , is plotted as a function of this ratio. Also the dynamical phase  $\phi_{11}$  is shown. One can note that the error is approximately null in a range of low detuning values. It increases monotonically as  $\Delta/\Omega$  is growing. Again, if the system is initially in  $|01\rangle$  (or  $|10\rangle$ ), the error  $\epsilon$  remains approximately null. Instead, as before, the dynamical phase  $\phi_{01}$  is strongly influenced by this ratio as already shown in Fig. 8a.



**FIG. 9** In blue, the error computed as in Eq. (23) for a system initially in state  $|11\rangle$  is plotted as a function of  $\Omega\tau$  and  $\Delta/\Omega$  in 9a and 9b respectively. In red, also the dynamical phase  $\phi_{11}$  acquired by the state  $|11\rangle$  after the two pulses is reported.

**b. Imperfect blockade.** As already explained in Sec. III, finite blockade interactions only affects the dynamics if the system is initially in the state  $|11\rangle$ . More specifically, in this case the dynamics is equivalent to that of a three-level system with states  $|a_2\rangle \equiv |11\rangle$ ,  $|b_2\rangle \equiv |1r\rangle + |r1\rangle$  and  $|c_2\rangle = |rr\rangle$  and can be described by the Hamiltonian in Eq. (14). In Fig. 10a the time-dependence of the population, for a pair of atoms initially in  $|11\rangle$  and for different values of the strength parameters  $V$ , is reported. One can note that, for low values of  $V$ , the system oscillates between the states  $|a_2\rangle$ ,  $|b_2\rangle$  and  $|c_2\rangle$  as expected. Instead, already for  $V = 100$ , the effect for finite blockade simply reduces to the two-level system with oscillations only between the states  $|a_2\rangle$  and  $|b_2\rangle$ . In Fig. 10b, the dynamical phases  $\phi_{01}$  and  $\phi_{11}$  are plotted as a function of  $\Delta/\Omega$  for  $V = 1$ . As the dynamics of the  $|01\rangle$  (or  $|10\rangle$ ) state is not affected by the finite blockade interactions, the phase  $\phi_{01}$  scales as already shown in Fig. 8a. On the other hand,  $\phi_{11}$  behaves differently and is characterized by several discontinuities. That is on account of phase computation which is periodic with a period of  $2\pi$ . In the plot 10b, we unwrap artificially the dynamical phase  $\phi_{11}$ . Hence, also in the case of finite blockade interactions, one can use  $\Delta$  as a control knob for the relative dynamical phases acquired by  $|11\rangle$  and  $|01\rangle$ , and thus realize a CZ gate.



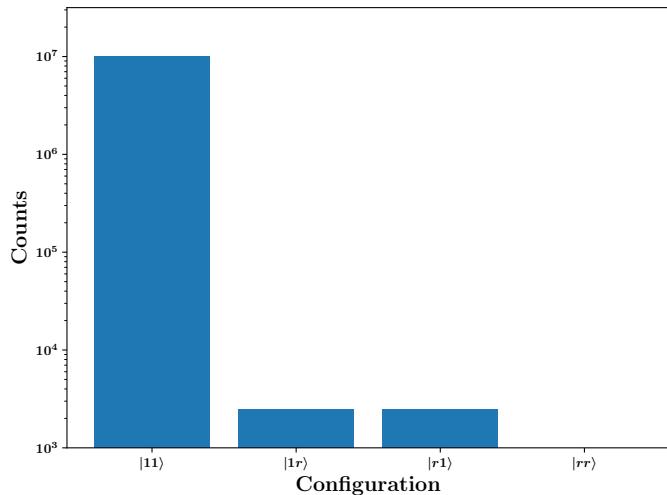
**FIG. 10** In 10a, the time-dependence of the population for the three-level system in Eq. (14) under finite blockade interactions assumption is reported. The parameters fixed for the simulation are the one in (13). In 10b, the dynamical phase of states in  $|01\rangle$  (or  $|10\rangle$ ) and  $|11\rangle$  is illustrated for  $V = 1$  as a function of the  $\Delta/\Omega$  ratio.

### B. Measurement process with Gaussian noise on $\tau$ and $\Delta$

Let us consider a pair of atoms initially in the state  $|11\rangle$ . Under the assumption of a random perturbation on the optimal parameters, we want to investigate what are the possible states allowed after the application of the CZ gate and with which probability they can be found in a measurement process.

To do this, we fix the optimal parameters as in (13) and we introduce a Gaussian noise with zero mean and a standard deviation of 1% on both the quantities  $\Omega\tau$  and  $\Delta/\Omega$ . Then, the CZ gate is applied and the wave-function of the final state is stored. The experiment is ran several times ( $\sim 10^3$ ) with these stochastic fluctuation. For each of the final states obtained, the measurement process is simulated: a random configuration is selected according to their probabilities. In particular, we take  $10^4$  measurements for each wave-function. All the results are grouped in order to obtain enough statistics.

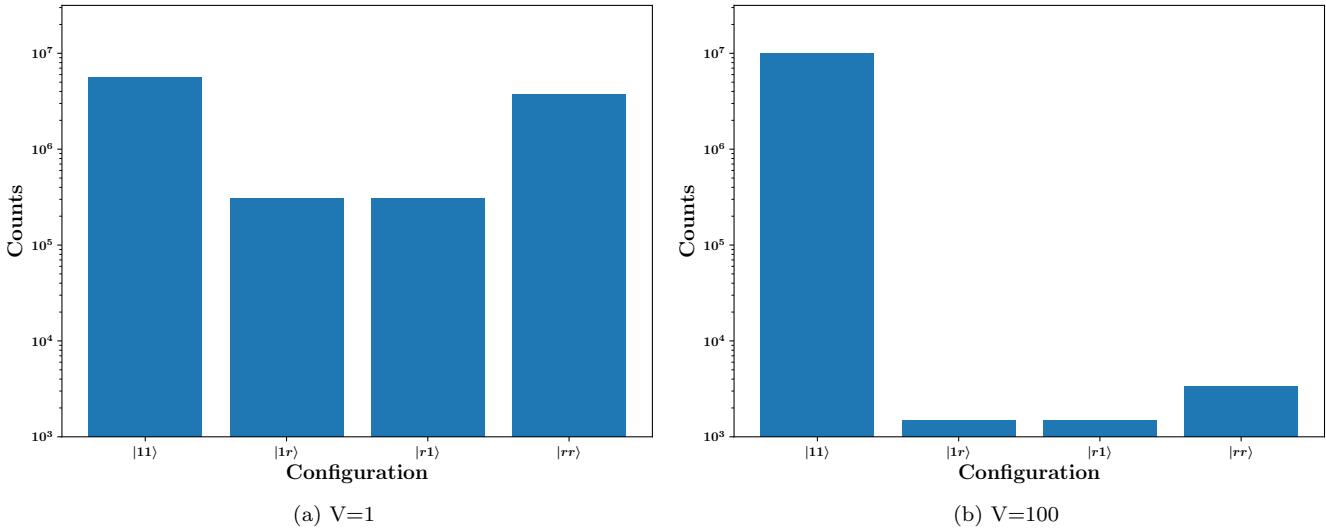
**a. Perfect blockade** If perfect blockade assumption holds, the dynamics of the system initially in  $|11\rangle$  is simulated according to the Hamiltonian in Eq. (5). In Fig. 11, we show the bar plot of the measurement process simulation explained above. The system with a random perturbation of 1% return with an high probability to its initial state  $|11\rangle$ . However, there is also a small probability to find the states  $|1r\rangle$  or  $|r1\rangle$ . One can also note that the final state  $|rr\rangle$  is not allowed as expected.



**FIG. 11** Bar plot of measurement process simulation for a system initially in  $|11\rangle$  under the application of the CZ gate in perfect blockade assumption. A Gaussian noise with mean zero and standard deviation of 1% is introduced on the quantities  $\Omega\tau$  and  $\Delta/\Omega$ . The experiment is ran  $10^3$  times and for each final wave-function  $10^4$  measurements are taken.

**b. Imperfect blockade.** Under the assumption of imperfect blockade, the system initially in  $|11\rangle$  evolves according to Eq. (14). In this case, for small values of the interaction strength parameter  $V > 0$ , we do not know the exact values of the optimal parameters to realize the CZ gate. However, by fixing the same parameters of (13), the measurement process is simulated and plotted in Fig 12a. It is interesting to note that now the state  $|rr\rangle$  is allowed. Indeed, for  $V = 1$  the states  $|11\rangle$  and  $|rr\rangle$  can be measured more or less with equal probability.

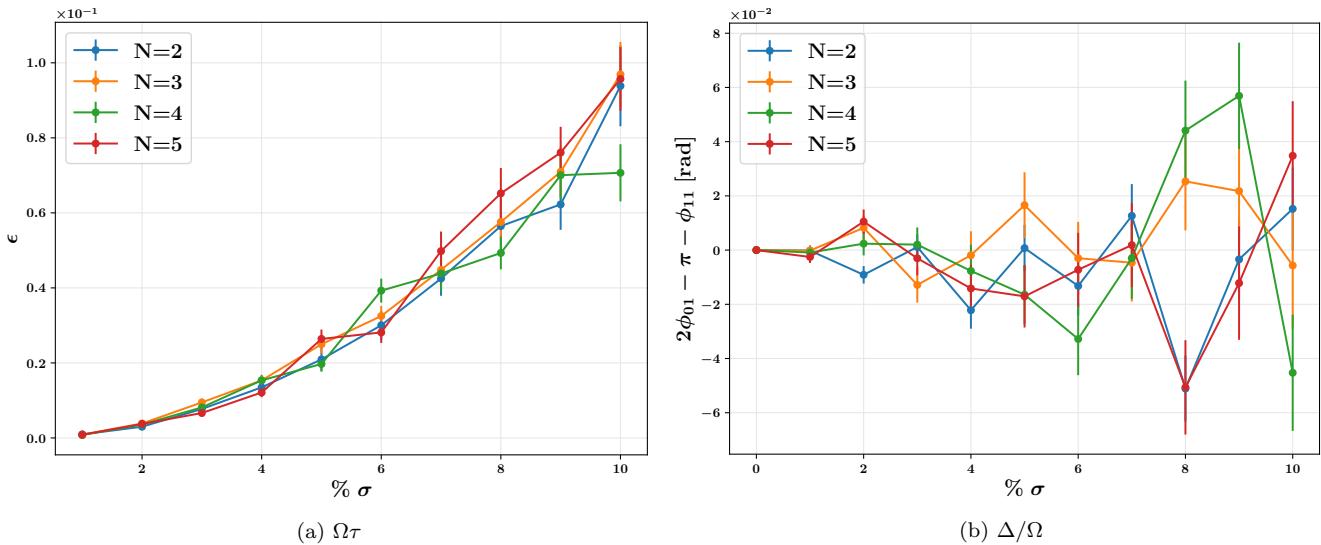
Now, let us consider the case  $V \gg |\Delta|, |\Omega|$ . As suggested by [1], for big values of the strength parameter the dynamic is equivalent to the one of a two level system providing that  $\Delta$  is renormalized by an amount  $\Omega^2/(2V)$ . In Fig. 12b, the measurement process is simulated for  $V = 100$ . One can note that, in this case the state  $|rr\rangle$  is allowed but with a very low probability, comparable to the one of states  $|1r\rangle$  and  $|r1\rangle$ . Hence, in the limit  $V \rightarrow \infty$  we expect that the counts of the  $|rr\rangle$  configuration becomes approximately null.



**FIG. 12** Bar plot of measurement process simulation for a system initially in  $|11\rangle$  under the application of the CZ gate in imperfect blockade assumption. A Gaussian noise with mean zero and standard deviation of 1% is introduced on the quantities  $\Omega\tau$  and  $\Delta/\Omega$ . The experiment is ran  $10^3$  times and for each final wave-function  $10^4$  measurements are taken.

## 2. CZ gate in a chain of N-qubit

Under the assumption of perfect blockade regime, we make a numerical simulation of a chain of  $N$ -qubit to investigate noise effects on the CZ gate implementation with an increasing number of qubits. In particular, the chain is implemented as explained in Sec. IV and it is initialized such that both the first and the last qubits are in state  $|1\rangle$ . A Gaussian noise with zero mean and standard deviation  $\sigma$  is introduced on the optimal parameters before the application of each pulse.



**FIG. 13** Noise effects on the CZ gate implementation in a  $N$ -qubit chain. A Gaussian noise with zero mean and standard deviation  $\sigma$  is introduced on  $\Omega\tau$  and  $\Delta/\Omega$  and the result is illustrated respectively in 13a and 13b. In particular,  $\% \sigma$  refers to the value of the standard deviation as a percentage of the optimal parameters. The error  $\epsilon$  in 13a and the phase difference in 13b are computed as the mean of 100 iterations.

**a. Gaussian noise on  $\Omega\tau$**  First of all, we study how a Gaussian noise on the duration of the pulse affects the dynamics. More specifically, we compute the error  $\epsilon$  of the final state (defined as in Eq. (23)) as a function of the noise percentage  $\% \sigma$  of the optimal parameter for  $\Omega\tau$ . In Fig. 13a, the results for a number of qubits up to 5 and

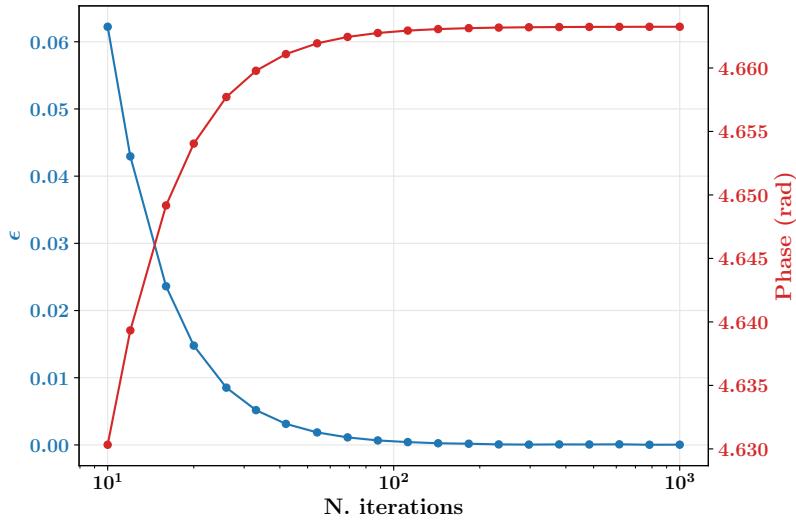
for a standard deviation up to 10% are shown. We note no significant trend for a different number of qubits in the chain. Instead, we observe that increasing the value of the standard deviation of the Gaussian noise the error of the final states increases super-linearly.

**b. Gaussian noise on  $\Delta/\Omega$**  Then, the Gaussian noise is introduced in the detuning parameter. As already discussed, fixing the optimal parameter  $\Omega\tau$  the state  $|11\rangle$  return to itself. Instead, by changing  $\Delta/\Omega$ , the phase of the state is affected. The CZ gate is correctly implemented if the relation  $2\phi_{01} - \pi - \phi_{11} = 0$  holds. So, in Fig. 13b we study how the noise affects this relation. Overall, one can note that a noise up to 10% does not affect significantly the dynamics. Indeed, the phase difference increases with the noise but still remains approximately null ( $\sim 10^{-2}$ ). As before, there is no significant trend increasing the number of qubits in the chain.

### 3. Timing analysis of time evolution methods

To compare the performances of different time evolution algorithms and of different implementations discussed in Sec. V, we perform a timing analysis. For this analysis, we set  $\Omega$ ,  $\Delta$  and  $\tau$  accordingly to optimal parameters in Eq. (13) and we use a 64-bit/8-byte complex type representation (4-byte for real part and 4-byte for imaginary part).

First of all, we analyze the error  $\epsilon$  of the state  $|11\rangle$  and its dynamical phase as a function of the number of iterations for Crank-Nicolson methods. To this purpose, the NumPy implementation with Crank-Nicolson method with LU decomposition is used. In Fig. 14, we observe that both error and phase curves stabilize to the expected values for a number of iterations above about 100. Analogous results are obtained for other implementations and methods. Therefore, we choose 100 iterations as optimal value to obtain shorter running time and accurate results.



**FIG. 14** Phase and error of state  $|11\rangle$  as a function of the number of iterations for two-qubits CZ gate. We consider NumPy implementation with Crank-Nicolson with LU decomposition method.

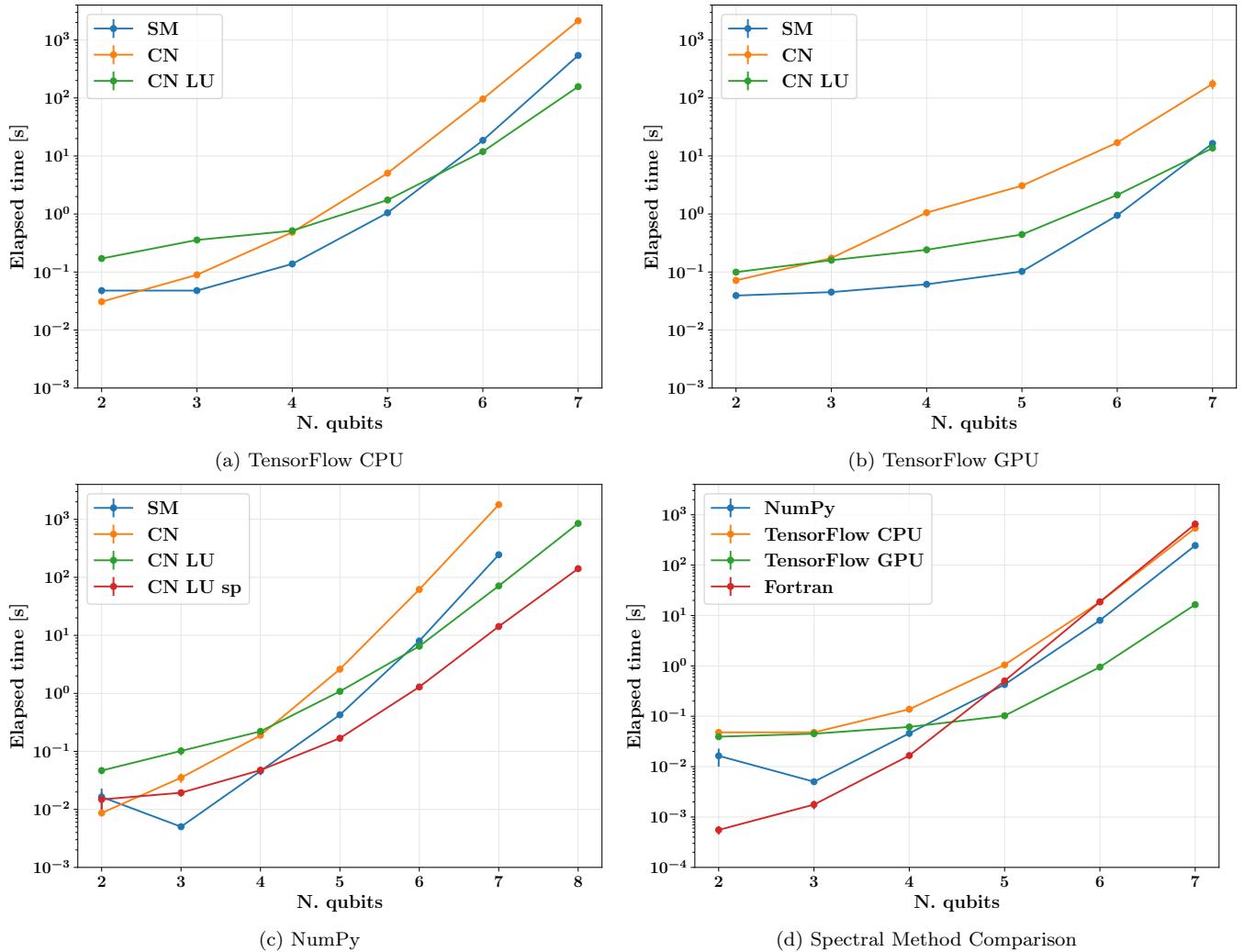
Then, the elapsed time for all methods for NumPy implementation and for TensorFlow implementation (both on CPU and GPU) is measured as a function of the number of qubits  $N$  in the chain. We compute the mean time over 10 realizations and we use JobLib Python library to parallelize the execution over multiple core of the CPU.

In Fig. 15a, 15b and 15c, we observe that in general Spectral Method (SM) have good performances for a small number of qubits, comparable with the Crank-Nicolson algorithm. However, its behavior highlight the fact that for a larger number of qubits the diagonalization of big matrices becomes soon unfeasible. Moreover, one can note that Crank-Nicolson algorithm with LU decomposition (CN LU) scales slower with respect to Crank-Nicolson algorithm (CN), as anticipated in Sec. V.3. Furthermore, in Fig. 15c, we see that sparse matrix optimization guarantees a gain in performance.

Observing Fig. 15a and 15b, one can note that, for a small number of qubits, TensorFlow implementation performs similarly on CPU and GPU for all methods. On the other hand, for a larger number of qubits the computation is significantly faster on GPU. In particular, for  $N = 7$ , the gain in elapsed time is of one order of magnitude. Moreover, the spectral method, which is based on eigenvalues decomposition and matrix inversion, is the one that mostly take

the advantage in speed with respect to its analogous on CPU. However, the amount of memory generally available for GPUs limits the length of the chain that can be simulated using such devices. In our implementation, the GPU we used run out of memory with a chain of  $N = 8$  qubits, trying to handle a  $3^8 \times 3^8$  matrix.

Finally, in Fig. 15d the results with spectral method for NumPy, TensorFlow and Fortran implementations are compared. We observe that for small number of qubits Fortran implementation significantly outperforms all the others, while for larger  $N$  it scales similarly to implementations ran on CPU. NumPy implementation scales as TensorFlow run on CPU for all number of qubits considered and results to be slightly faster. In conclusion, tensorflow implementation on GPU outperforms other methods for a large number of qubits.



**FIG. 15** Mean elapsed time in seconds as a function of the number of qubits in the chain. Spectral Method (SM), Crank-Nicolson method (CN), Crank-Nicolson with LU decomposition method (CN LU) and Crank-Nicolson with LU decomposition with sparse matrix method (CN LU sp) are compared. We consider TensorFlow implementation on CPU (15a), TensorFlow implementation on GPU (15b) NumPy implementation (15c) and a comparison between all implementations with Spectral methods (15d).

## VII. CONCLUSIONS

In this Report, we simulate numerically the dynamics of a two-qubit system under the application of a controlled-phase gate.

First of all, we check the correctness of the numerical implementation by fixing the optimal parameters suggested and comparing the results with the one of [1].

Then, we vary the optimal parameters in order to investigate the behavior of the CZ gate in response to this variation. We see that a proper choice of  $\Omega\tau$  and  $\Delta/\Omega$  is needed so that a state initially in  $|11\rangle$  return to itself after the CZ gate application. Instead, if the system is in  $|01\rangle$  (or  $|10\rangle$ ), it return to itself independently on the optimal choice of the parameters  $\Omega\tau$  and  $\Delta/\Omega$ , but a proper choice of the phase jump between the two pulses is needed. The dynamical phase acquired both by  $|11\rangle$  and  $|01\rangle$  (or  $|10\rangle$ ) depends strongly on this choice. This is a crucial point for a correct realization of a controlled-phase gate for which  $\phi_{11} = 2\phi_{01} - \pi$ .

Moreover, a Gaussian noise is also introduced in the system and the measurement process is simulated, both in perfect and imperfect blockade regime, obtaining the expected results. In the last case, we note that increasing the strength parameter  $V$ , the state  $|rr\rangle$  less and less likely.

After that, the CZ gate is implemented in a chain of  $N$ -qubit and also in this case a Gaussian noise before each pulse is provided. We perturb separately the parameter  $\Omega\tau$  and  $\Delta/\Omega$  and we monitor respectively the error  $\epsilon$  and the relation  $\phi_{11} = 2\phi_{01} - \pi$ . In both cases, we do not observe any significant difference as a function of the standard deviation of the normal noise for different number of qubits.

Finally, a timing analysis of unitary time evolution algorithms and of their different implementations is performed. We conclude that for a lower number of qubits the spectral method have good performances especially in the TensorFlow GPU implementation. However, this method is not feasible for a larger number of qubits. Indeed, approximation are needed in order to handle such big matrices. One can rely on Crank-Nicolson algorithm with LU decomposition which for large matrices seems to scale slowly. Even better, it is the NumPy implementation of Crank-Nicolson algorithm with LU decomposition with sparse matrices.

All the implemented code and the results shown in this work are available on GitHub [4].

## REFERENCES

- [1] Harry Levine, Alexander Keesling, Giulia Semeghini, Ahmed Omran, Tout T. Wang, Sepehr Ebadi, Hannes Bernien, Markus Greiner, Vlatko Vuletić, Hannes Pichler, and Mikhail D. Lukin. Parallel implementation of high-fidelity multiqubit gates with neutral atoms. *Phys. Rev. Lett.*, 123:170503, Oct 2019.
- [2] Peter Lambropoulos and David Petrosyan. *Fundamentals of Quantum Optics and Quantum Information*. Springer-Verlag, Berlin, Heidelberg, 2006.
- [3] Mark Fox. *Quantum optics: an introduction*. Oxford master series in atomic, optical, and laser physics. Oxford Univ. Press, Oxford, 2006.
- [4] Alice Pagano and Michele Puppin. Two-qubit CZ gate implementation with trapped neutral atoms:a numerical simulation, year = 2021, url = <https://github.com>, urldate = 03-03-2021.