Coexisting Raman-Scattering Depolarization Simulation

Overview

This code simulates quantum communication experiments utilizing coexistence with classical signals, such as in fiber optic links carrying both quantum and classical data. This is modelled using the depolarization channel.

For a complete theoretical explanation of the methods employed, read the associated pre-print: "*Simulating Raman Scattering Impairments with Depolarization Noise in Quantum-Classical Links"* [1].

The main entry point is:

`run_raman_depolar_sim.py`

This script allows the user to run 3 types of coexistence experiments:

1. **Direct Transmission** (`import_coexisting_direct_transmission`)
2. **Entanglement Distribution** (`import_coexisting_entanglement`)
3. **Teleportation** (`import_coexisting_teleportation`)

Both simulate how **Raman noise**, modeled as **depolarization**, affects quantum fidelity, based on input **visibility** values.

Guidance on Usage

There are two main use cases for this simulator:

1. **Estimating the coexisting fidelity value of a coexisting link configuration (fibre + detector setup).** The user may be interested in comparing the fidelity difference of two hardware configurations (e.g. wavelength channel selections or launch powers and their respective Raman photons received at the detector), perhaps with respect to a minimum fidelity requirement of the protocol. In this case, the simulator requires no modification and input parameters can be tweaked to output the coexisting fidelity.
2. **Simulating larger network protocols in NetSquid with coexisting links.** If the user is intending to include coexisting noise in link or network layer protocol simulation, this

simulator can be forked, or subsections utilized, to apply the appropriate coexisting noise to the transmitted qubits to yield the appropriate noisy coexisting density matrix.

Both of these cases will follow a similar workflow:

1. **Provided by the user in the API**: Determine the number of incident Raman photons at the detector. This is from experimental measurements, or from an estimated number.
2. **Performed by the simulator:** Calculates the mixing probability of the link based on the incident Raman photons. Applies depolarization with this mixing probability to NetSquid qubit objects.
    a. Note: for teleportation using a midpoint Bell-state Measurement, there will be corresponding mixing probabilities for both fibres utilized (Alice's qubit to be teleported → BSM, Bob's entangled signal photon → BSM)
3. **Simulator returns depolarized fidelity and depolarized qubit(s).**

See below for more tailored workflows for each scenario.

---

1. `run_raman_depolar_sim.py`

Purpose

This is the main runner file that executes the simulation. It lets you specify:

- Type of experiment
- Hardware parameters
- Optional flags (e.g. verbosity)

Execution

- `python run_raman_depolar_sim.py`
- Modify the file directly to change input parameters or flags.

Output

- Printed fidelity results for each visibility values, and depolarized qubit(s)

---

2. Interfacing with `import_coexisting_direct_transmission.py`

Purpose

Simulates coexisting **direct quantum transmission** of a single pure qubit from an emitter to a receiver over a fiber with **depolarizing noise**.

Key Functions

- `run_coex_direct_experiment(noisy_visibility, random_seed=1, input_state='0', verbose=False)`
    - Converts incident Raman photons to depolarizing probability.
    - Sets up a network with one coexisting quantum channel.
    - Sends a single pure qubit and retrieves the noisy output.
    - Calculates fidelity of the output vs. the ideal state.
    - Returns: (depolarized fidelity, depolarized state)

Usage

Within `run_raman_depolar_sim…`

- Determine the number of incident Raman photons at the detector per second
- Input this value and other hardware parameters into `run_coex_direct_transm_experiment()` to run the experiment.

---

3. Interfacing with `import_coexisting_entanglement.py`

Purpose

Simulates coexisting **entanglement distribution** through a noisy fiber link. Uses a pure Bell-pair source and characterizes the received state fidelity.

Key Functions

- `run_coex_ent_experiment(noisy_visibility, random_seed=1, bell_state="phi+", verbose=True)`
  - Converts visibility to depolarizing probability.
  - One half of the Bell pair is stored locally; the other travels through the depolarizing fiber.
  - Fidelity is computed between the received (noisy) Bell state and the original.
  - Returns: (depolarized fidelity, depolarized entangled qubit 1, depolarized entangled qubit 2)

Usage

Within `run_raman_depolar_sim…`

- Determine the number of incident Raman photons at the detector
- Calculate the noisy visibility of the signal photon's fibre based on the number of incident Raman photons at the detector and other hardware parameters listed in the parameter dictionary.
  - Using `calc_visibility()` from `import_coexisting_teleportation`
  - **Note**: skip these first 2 steps if noisy visibility is already known
- Input noisy visibility to `run_coex_ent_experiment()` to run experiment

---

3. `import_coexisting_teleportation.py`

Purpose

Simulates coexisting **teleportation** through a noisy fiber link. Alice prepares a pure state to teleport. Bob prepares the pure entangled Bell-state. Bob keeps the entangled idler. Both send their signal photons to a midpoint optical Bell-state measurement. Both experience (possibly unique) coexisting noise, with a corresponding amount of detected Raman photons within the BSM component and idler receivers. See [2] for the full setup.

Key Functions

- `run_coex_ent_experiment(noisy_visibility, random_seed=1, bell_state="phi+", verbose=True)`
    - Converts visibility to depolarizing probability.
    - Fidelity of teleportation is computed as function of Alice's qubit's fidelity and the entangled fidelity after coexisting transmission.
    - Returns: (depolarized fidelity, depolarized entangled qubit 1, depolarized entangled qubit 2, depolarized Alice's qubit)

Usage

Within `run_raman_depolar_sim…`

- Determine the number of incident Raman photons at the detector
- Calculate the noisy visibility of each signal photon's fibre based on the number of incident Raman photons at the detector and other hardware parameters listed in the parameter dictionary.
    - Using `calc_visibility()` from `import_coexisting_teleportation`
    - For teleportation using a midpoint Bell-state Measurement, there will be corresponding mixing probabilities for both fibres utilized (Alice's qubit to be teleported → BSM, Bob's entangled signal photon → BSM)
    - Note: skip these first 2 steps if noisy visibility is already known
- Input noisy visibility to `run_coex_tele_experiment()` to run experiment

Dependencies

- `NetSquid` for quantum simulation components (nodes, protocols, channels)

- `pydynaa` for event handling

- `matplotlib` and `pandas` (optional, for plotting and data handling)

References

1. Smith, J., & Proietti, R. Simulating Raman Scattering Impairments with Depolarization Noise in Quantum-Classical Links. Photonics in Switching and Computing (2025). To Appear.
2. Thomas, J. M., Yeh, F. I., Chen, J. H., Mambretti, J. J., Kohlert, S. J., Kanter, G. S., & Kumar, P. (2024). Quantum teleportation coexisting with classical communications in optical fiber. Optica, 11(12), 1700-1707.