Estimation of Spin Observable using Particle filtering

Kaumudibikash Goswami[1,*]

[1]QICI Quantum Information and Computation Initiative, School of Computing and Data Science, The University of Hong Kong, Pokfulam Road, Hong Kong.

I. PROBLEM STATEMENT

In this work, we consider the following scenario. An experimentalist has a preparation device that creates many copies of an initial quantum state, e.g., the ground state of a two-level system. Due to internal noise in the device, the initial setting suffers from random noise, resulting in each initial preparation being slightly off from the other. An environmental magnetic field and an oscillating magnetic field by the experimentalist characterize the Hamiltonian, resulting in the evolution of the state. This results in an oscillatory behaviour in the expectation value of the spin observables, where the oscillation frequency is given by the Rabi frequency. The experimentalist aims to track the changing value of this spin observable by creating many copies of the states and measuring the state in the relevant Pauli basis. However, motivated by Example 7.2 of Ref. [1], we introduce two sources of noise in the measurement: sometimes the measurement device randomly malfunctions and gives a measurement outcome sampled from a uniform distribution; in contrast, in the case of a 'valid' measurement, the data is riddled with Gaussian noise. This introduces significant outliers that deviate substantially from the true signal, creating a challenging dataset. Despite these obstacles, the experimentalist proceeds to estimate the spin observable using a Particle filtering algorithm, specifically the Bootstrap filtering algorithm.

In Sec. II, we first intuitively explain the algorithm, then in Sec. III, we give a detailed description of how the algorithm is customized for that specific problem, and the Hamiltonian, and the measurement noise. Then in Sec. IV, we discuss the details of the customized algorithms and results, and finally, in Sec. V, we summarise our work.

II. PARTICLE FILTERING

Suppose we have a dynamically evolving state $x_{0:k}$ where x_k is the state at the k-th time step. At each time step k, we perform a measurement on the state to obtain the data y_k. The task is to estimate some properties of the state, given by $f(x_k)$, from the obtained measurement dataset $y_{1:k} := \{y_1, y_2, \cdots, y_k\}$. We assume the underlying dynamics is Markovian, i.e., the probability the state at a particular time steps depends only on the state at the previous time step: $P(x_k|x_{0:k-1}, y_{0:k-1}) = P(x_k|x_{K-1})$ and the measurement result at a certain time step depends only on the state at that time step: $P(y_k|x_{0:k}, y_{0:k-1}) = P(y_k|x_K)$. Let us now describe the particle filtering algorithm via the following detective story.

- **Initialization:** Imagine Sherlock Holmes is in search of a famous historical artifact. To gather the necessary information, he sends out his 'Baker Street Irregulars' all over the city—some in the library, some in the city tower.

  This is the initialization step of the algorithm, where the algorithm generates a set of N samples $\{x_0^{(i)}\}_{i=1}^N$, each representing a possible state (e.g., a position or parameter value) drawn from an initial probability distribution $p(x_0)$.

- **State update:** As Sherlock finds clues, like a dusty footprint near the library or a faint glow from the tower, he updates his initial belief about the location of the artifact. Note that this updated belief incorporates the underlying state-space dynamics: a clever adversary can keep moving the artifact, possibly using a well-thought-out plan or simply flipping a coin. According to his updated belief, he reassigns his network.

  In the algorithm, this corresponds to the state update part. The algorithm generates new samples $\{x_k^{(i)}\}_{i=1}^N$ from the updated distribution, $p(x_k|x_{k-1}^{(i)})$. This update includes the underlying state-space dynamics along with the process noise. For example, if the state-space model is governed by a $x_k = f(x_{k-1}) + \mathbf{n}_k$ where $\mathbf{n}_k$ is a sample from a normal distribution of mean 0 and covariance matrix Q: $\mathbf{n}_k \sim \mathcal{N}(0, Q)$, then $p(x_k|x_{k-1}^{(i)}) = \mathcal{N}(f(x_{k-1}^{(i)}), Q)$.

- **Weight update and Resampling:** It is important for Sherlock to judiciously use his human resources. So he gives more 'weight' to the places where he finds important clues, e.g., the library or tower. Accordingly, he sends more of his network to those places, and fewer to unlikely spots like the local park. Note that during this process, Sherlock has to deal with

* goswami.kaumudibikash@gmail.com

the noise and distractions. Sometimes, the clues get messy: maybe a gust of wind moves the footprint, or a candle flickers to mislead them. This is like "noise" or "clutter" in the data. Sherlock uses his smarts to weigh the clues, keeping his networks focused on the most reliable hints.

In the algorithm, this corresponds to the weight update and resampling steps. The algorithm assigns each sample's weight $w_k^{(i)}$ based on how well it matches the new measurement y_k (the clue), using the likelihood $p(y_k|x_k^{(i)})$. Evaluation of the likelihood incorporates the measurement noise: if $\mathbf{y}_k = g(x_{k-1}) + \mathbf{r}_k$ where $\mathbf{r}_k \sim \mathcal{N}(0, R)$, then $p(y_k|x_k^{(i)}) = \mathcal{N}(g(x_k^{(i)}), R)$, and $w_k^{(i)} \propto p(y_k|x_k^{(i)})$.

Resampling then discards low-weight particles and duplicates high-weight ones, concentrating the particles where the evidence (clues) is strongest. This is done by generating N indices $\{i\}_{i=1^N}$ according the discrete probability mass function $\{w_k^i\}_i$ (after normalisation) and retaining samples $x_k^{(i)}$.

- **Final outcome:** After exploring more clues, like an old map fragment, most of the networks cluster around a secret room behind the tower. Sherlock notices this pattern and trusts the "weight" of the evidence. This is how particle filtering hones in on the best guess by boosting the important particles.

A. Algorithmic structure

The following algorithm describes the particle filtering process described above.

Algorithm 1.
- **Input**: Initial distribution $p(x_0)$, number of samples N, time step Δt, total time T.
- **Output**: Estimated state $\hat{x}_k$ at each time step k
- **Initialization**:
  - For each index i from 1 to N:
    - Draw initial samples $x_0^{(i)}$ from $p(x_0)$
  - Set initial weights $w_0^{(i)} = 1/N$ for all i
- For each time step k from 1 to T_N=$T/\Delta t$:
  - **State update**:
    - For each index i from 1 to N:
      - Predict new particle $x_k^{(i)}$ from $p(x_k|x_{k-1}^{(i)})$
  - **Weight Update and Resampling**:
    - For each index i from 1 to N:
      - Compute weight $w_k^{(i)} \propto p(y_k|x_k^{(i)})$
      - Normalize weights: $w_k^{(i)} \leftarrow w_k^{(i)} / \sum_{j=1}^N w_k^{(j)}$
    - Resample N indices $\{i_1, i_2, \ldots, i_N\}$ according to the discrete probability mass function $\{w_k^{(i)}\}_i$
    - Set $x_k^{(i)} \leftarrow x_k^{(i_j)}$ and $w_k^{(i)} \leftarrow 1/N$ for i from 1 to N
  - **State Estimation**:
    - Compute estimated state $\hat{x}_k = \sum_{i=1}^N w_k^{(i)} x_k^{(i)}$

III. PARTICLE FILTERING FOR QUANTUM STATES

Note that in the conventional particle filtering algorithm state space is considered to be classical. In this work, we aim to introduce a variant of the particle filtering algorithm where the underlying state space is governed by quantum theory, i.e., an element of a complex vector space. In particular, here we consider a two-level quantum system exposed to a controlled magnetic field, which results in Rabi oscillation in the state's spin observable. We will discuss the detailed Hamiltonian model soon. The dynamics of the state is given by Schrödinger's equation. Suppose at times steps k and $k-1$, the quantum states are $|\psi_k\rangle$, and $|\psi_{k-1}\rangle$, and the underlying Hamiltonian is $\tilde{H}$ (time-independent in the rotating frame), then the solution of Schrödinger's equation for a discrete time duration Δt gives us

$$|\psi_k\rangle = \exp\Big[-i\tilde{H}\Delta t\Big] |\psi_{k-1}\rangle = U_{\Delta t} |\psi_{k-1}\rangle \,. \tag{1}$$

Note that one can map this dynamics to a classical state space dynamics by defining a classical state $x_k := \langle\sigma_x\rangle_{\psi_k} = \langle\psi_k|\sigma_x|\psi_k\rangle$, where σ_x is the Pauli-x operator. This allows us to consider a deterministic state space model

$$\langle\sigma_x\rangle_{\psi_k} = x_k = g(x_{k-1}) = \langle\sigma_x\rangle_{U_{\Delta t}\psi_{k-1}} \tag{2}$$

$$P(x_k|x_{k-1}) = \delta_{x_k,\ g(x_{k-1})}. \tag{3}$$

Here, $\delta(\cdot,\cdot)$ is the Kronecker delta function. Here, while we have considered the state update rule to be deterministic, in principle, the work can be extended to the Stochastic Schrödinger equation. Physically, such a scenario can appear when the experimentalist's controlled magnetic field suffers from some noise, for example, the magnetic field can experience some Brownian motion due to a nearby Microwave. Although the state update rule is deterministic, we allow for noisy initial state preparations. We describe the noise model in the algorithm. Finally, the experimentalist's measurement device also suffers from noise, which we discuss in the next section.

A. Simulation of the measurement noise

The measurement process for the spin observable σ_x is designed to incorporate both stochastic noise and systematic errors to emulate realistic experimental imperfections. The simulation procedure is as follows: the true expectation value of the spin observable, $\langle\sigma_x\rangle_{\psi_k}$, is calculated from the quantum state evolved in the rotating frame under the time-independent Hamiltonian $\tilde{H}$. To simulate measurement noise encountered in practical quantum experiments, a Gaussian noise term with zero mean and variance $R = 0.1$ is added to it, reflecting small, random fluctuations in the measurement apparatus. Additionally, to model severe measurement faults, such as those arising from equipment malfunctions, calibration errors, or external interference, each measurement has a corruption probability $c_p = 0.5$ of being replaced by a random value drawn from a uniform distribution over $[-2, 2]$. The combination of Gaussian noise and probabilistic corruption ensures that the simulated measurements capture both subtle inaccuracies and extreme errors, providing a realistic testbed for estimating the true spin observable $\langle\sigma_x\rangle_{\psi_k}$ using Bayesian filtering methods.

B. Hamiltonian model

Here we provide the continuous time version of the quantum state evolution. We explore the well-studied phenomena of *Rabi cycle* [2]. Consider an experimentalist attempting to interact with a two-level spin-1/2 system by subjecting it to a magnetic field. The overall magnetic field $\vec{B}$ can be decomposed as follows: the fixed environmental magnetic field, $\vec{B_0}$ along the z-direction, and the experimentalist's oscillating magnetic field $\vec{B}_{\text{exp}}$ with the oscillation being restricted to the $x-y$ plane:

$$\begin{aligned}\vec{B} &= \vec{B_0} + \vec{B}_{\text{exp}} \\ &= |B_0|\hat{z} + \left|B_{\text{exp}}\right|\cos\left(\omega_f t\right)\hat{x} + \left|B_{\text{exp}}\right|\sin\left(\omega_f t\right)\hat{y}.\end{aligned} \tag{4}$$

Here, $\hat{x}$, $\hat{y}$ and $\hat{z}$ are the unit vectors along the x, y and z directions, $|B_0|$ and $\left|B_{\text{exp}}\right|$ are the magnitudes of the $\vec{B_0}$ and $\vec{B}_{\text{exp}}$, respectively and ω_f is the angular velocity of the oscillation.

To obtain the corresponding Hamiltonian, considering the magnetic moment $\vec{\mu} = -\gamma_e\vec{\sigma}/2$ of the spin-1/2 particle, where γ_e being the gyromagnetic ratio, and $\vec{\sigma} := \sigma_x\hat{x} + \sigma_y\hat{y} + \sigma_z\hat{z}$ being the Pauli observables:

$$\sigma_x = \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}, \sigma_y = \begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}, \sigma_z = \begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}. \tag{5}$$

We can express the effective Hamiltonian as

$$\begin{aligned}H(t) &= -\vec{\mu}.\vec{B} \\ &= \frac{\gamma_e|B_0|}{2}\sigma_z + \frac{\gamma_e\left|B_{\text{exp}}\right|}{2}[\cos\left(\omega_f t\right)\sigma_x + \sin\left(\omega_f t\right)\sigma_y].\end{aligned} \tag{6}$$

We now define $\omega_0 := \gamma_e|B_0|$, and $\omega_c = \gamma_e\left|B_{\text{exp}}\right|$. Accordingly, we have the following Hamiltonian:

$$H(t) = \frac{1}{2}\begin{bmatrix}\omega_0 & \omega_c e^{-i\omega_f t}\\ \omega_c e^{-i\omega_f t} & -\omega_0.\end{bmatrix} \tag{7}$$

In general, solving the corresponding Schrödinger equation through a time-dependent Hamiltonian requires solving the time-ordered exponential or approximating the solution via the method of perturbations. However, in this case, the evolution can be efficiently solved using the so-called 'rotating frame transformation': we first move to the rotating frame of oscillation to have an effective time-independent Hamiltonian, solve the Schrödinger equation, and then transform the final state back to the lab reference frame. More precisely, given the experimentalist's state $|\psi(t)\rangle$, we define the state in the rotating frame to be $\left|\tilde{\psi}(t)\right\rangle := U(t)^\dagger |\psi(t)\rangle$, with $U(t) := \exp\left[-i\omega_f t\sigma_z/2\right]$. One can see that the effective Hamiltonian $\tilde{H}$ in the rotating frame becomes time-independent:

$$\tilde{H} := U(t)^\dagger H(t)U(t) - iU(t)^\dagger \frac{dU(t)}{dt} \tag{8}$$

$$= \frac{1}{2}\begin{bmatrix} \Delta\omega & \omega_c \\ \omega_c & -\Delta\omega \end{bmatrix}, \qquad \text{where } \Delta\omega := \omega_0 - \omega_f. \tag{9}$$

Eigendecomposition of $\tilde{H} = \sum_{k\in\{+,-\}} \lambda_k |v_k\rangle\langle v_k|$ yields

$$\lambda_\pm = \pm\frac{\omega_{\text{eff}}}{2} \qquad \text{with } \omega_{\text{eff}} = \sqrt{\omega_c^2 + \Delta\omega^2} \tag{10}$$

$$|v_+\rangle = \cos\frac{\theta}{2}|0\rangle + \sin\frac{\theta}{2}|1\rangle, \tag{11}$$

$$|v_-\rangle = \sin\frac{\theta}{2}|0\rangle - \cos\frac{\theta}{2}|1\rangle, \quad \text{where } \theta = \arctan\left(\frac{\omega_c}{\Delta\omega}\right), \tag{12}$$

which allows us to figure out the evolved state $|\psi(t)\rangle$ in the lab frame:

$$|\psi(t)\rangle = U(t)\left|\tilde{\psi}(t)\right\rangle = U(t)\exp\left[-i\tilde{H}t\right]\left|\tilde{\psi}(0)\right\rangle \tag{13}$$

$$= U(t)\sum_{k\in\{+,-\}} \exp[-i\lambda_k t]\,\langle v_k|\psi(0)\rangle\,|v_k\rangle \tag{14}$$

Here the third equality is because $\left|\tilde{\psi}(0)\right\rangle = |\psi(0)\rangle$.

A well-studied observation is that for $|\psi(0)\rangle = |0\rangle$, in the rotating frame, the expectation value of the spin observable σ_x oscillates with time at the Rabi frequency $2\pi/\omega_{\text{eff}}$, the amplitude of the oscillation depends on $\Delta\omega$, specifically when the resonance is achieved, $\Delta\omega = 0$, the oscillation completely vanishes. More precisely, we have

$$\langle\sigma_x\rangle_{\tilde{\psi}(t)} = \langle\tilde{\psi}(t) \mid \sigma_x \mid \tilde{\psi}(t)\rangle = \frac{2\Delta\omega\,\omega_c}{\omega_{\text{eff}}^2}\sin^2\left(\frac{\omega_{\text{eff}}t}{2}\right) \tag{15}$$

In this work, we aim to estimate $\langle\sigma_x\rangle_{\tilde{\psi}(t)}$ using Bayesian filtering. In the next section, we now introduce all the algorithms customized for this problem and the results.

IV. ALGORITHMS AND RESULTS

A. All algorithms

Below, we describe how data is generated based on the above Hamiltonian model.

Algorithm 2 (Simulation of Faulty Spin Observable Data for Rabi Cycle)**.** This algorithm simulates the evolution of a two-level spin-1/2 system under a time-independent Hamiltonian in the rotating frame, computes the expectation value of the spin observable σ_x, and introduces measurement noise and faulty data for Bayesian filtering analysis.

- **Initialize parameters and matrices:**
  - Set the time step $\Delta t = 0.01$.
  - Define Larmor frequency $\omega_0 = 2\pi$ rad/s, control field strength $\omega_c = 0.5\omega_0$, and oscillation frequency $\omega_f = 0.8\omega_0$.
  - Compute $\Delta\omega = \omega_0 - \omega_f$ and effective frequency $\omega_{\text{eff}} = \sqrt{\omega_c^2 + \Delta\omega^2}$.
  - Construct the time-independent Hamiltonian in the rotating frame: $\tilde{H} = \frac{1}{2}\begin{bmatrix} \Delta\omega & \omega_c \\ \omega_c & -\Delta\omega \end{bmatrix}$.

  - Compute the time evolution operator: $U_{\Delta t} = \exp\left(-i\tilde{H}\Delta t\right)$.
  - Set measurement noise variance $R = 0.1$ and number of time steps $T_N = 500$.
  - Initialize the quantum state $|\psi(0)\rangle = |0\rangle = \begin{bmatrix}1\\0\end{bmatrix}$.
  - Set corruption probability $c_p = 0.5$ for faulty measurements.

- **Simulate time evolution and measurements:**
  - Initialize arrays for time T, states X, noisy measurements Y, and true measurements Y_{st}.
  - Set initial time $t = 0$ and state $|x\rangle = |\psi(0)\rangle$.
  - For $k = 1$ to T_N:
    * Update state: $|x\rangle \leftarrow U_{\Delta t}|x\rangle$.
    * Compute true spin observable: $y_{\text{st}} = \langle x|\sigma_x|x\rangle$.
    * Generate noisy measurement: $y = y_{\text{st}} + \sqrt{R}\cdot\mathcal{N}(0,1)$, where $\mathcal{N}(0,1)$ is a standard normal random variable.
    * Update time: $t \leftarrow t + \Delta t$.
    * Store t, $|x\rangle$, y_{st}, and y in T, X, Y_{st}, and Y, respectively.

- **Introduce faulty measurements:**
  - For each measurement y in Y, with probability c_p, replace y with a random value uniformly distributed in $[-2, 2]$.

- **Compute theoretical prediction:**
  - For each time t in T, compute the theoretical expectation value of σ_x: $\langle\sigma_x\rangle_{\tilde{\psi}(t)} = \frac{2\Delta\omega\omega_c}{\omega_{\text{eff}}^2}\sin^2\left(\frac{\omega_{\text{eff}}t}{2}\right)$.

- **Output:**
  - Return time array T, state array X, noisy measurements Y, true measurements Y_{st}, and theoretical predictions.

We now use this dataset in the particle filter algorithm. The algorithm estimates the spin observable σ_x for a two-level spin-1/2 system by tracking the quantum state evolution under noisy and faulty measurements. It initializes a set of N=10,000 samples, each representing a possible quantum state, sampled around the initial state $|0\rangle$ with random phases drawn from a Gaussian distribution with mean 0 and variance 0.1. At each time step, the algorithm propagates samples using the unitary time evolution operator $U_{\Delta t}$, derived from the rotating frame Hamiltonian $\tilde{H}$. Particle weights are updated based on the likelihood of observed measurements, which include Gaussian noise (variance $R = 0.1$) and random faults (uniformly distributed in $[-2, 2]$ with probability $1 - c_p = 0.5$). Resampling focuses on high-probability particles, and the mean of the observable σ_x is computed from the particles' expectation values. The algorithm outputs these estimates and calculates the root mean square error (RMSE) to quantify performance against theoretical predictions, effectively handling the non-linear and non-Gaussian measurement model.

Algorithm 3 (Particle Filter for Spin Observable Estimation)**.** This algorithm estimates the spin observable σ_x of a two-level spin-1/2 system using a particle filter to handle noisy and faulty measurements.

- **Initialize parameters:**
  - Set number of particles $N = 10{,}000$.
  - Load measurement data Y, corruption probability $c_p = 0.5$, measurement bounds $ub = 2$, $lb = -2$, and noise variance $R = 0.1$.
  - Define initial state $|\psi(0)\rangle = |0\rangle = \begin{bmatrix}1\\0\end{bmatrix}$ and Pauli matrix $\sigma_x = \begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$.
  - Set time evolution operator $U_{\Delta t} = \exp\left(-i\tilde{H}\Delta t\right)$, where $\tilde{H}$ is the Hamiltonian in the rotating frame.

- **Generate initial particle set:**
  - For $i = 1$ to N:
    * Sample initial angle setting $\Phi_i \sim \mathcal{N}(0, 0.05)$.
    * Set particle $|s_i\rangle = \begin{bmatrix}\cos(\Phi_i)\\\sin(\Phi_i)\end{bmatrix}$.

  - Store particles in set $SX = [|s_1\rangle, \ldots, |s_N\rangle]$.

- **Particle filtering loop:**
  - Initialize array MM to store mean estimates of σ_x.
  - For each time step $k = 1$ to $|Y|$:
    * Update particles and compute weights using Importance Sampling Bootstrap (Algorithm 4).
    * Resample particles using Resampling (Algorithm 5).
    * Compute estimated observable: $Y_{\text{est},i} = \langle s_i|\sigma_x|s_i\rangle$ for each particle $|s_i\rangle$.
    * Compute mean estimate: $m = \frac{1}{N}\sum_{i=1}^{N} Y_{\text{est},i}$.
    * Store m in $MM(k)$.

- **Output:**
  - Return mean estimates MM and plot against time T, measurements Y, and theoretical predictions.
  - Compute RMSE: $\sqrt{\frac{1}{|Y|}\sum_{k=1}^{|Y|}(\text{Theoretical pred}(k) - MM(k))^2}$.

Importance Sampling Bootstrap Algorithm: The Importance Sampling Bootstrap algorithm updates the particle set and their weights to reflect the likelihood of observed measurements in the particle filter framework. It propagates each particle using the unitary evolution operator $U_{\Delta t}$, computed from the rotating frame Hamiltonian $\tilde{H}$, to simulate the quantum state dynamics of the spin-1/2 system. The algorithm then evaluates the measurement likelihood, distinguishing between valid measurements, which follow a Gaussian distribution with variance $R = 0.1$, and faulty measurements, which are uniformly distributed in $[-2, 2]$ with probability $1 - c_p = 0.5$. For valid measurements, weights are computed using a Gaussian likelihood function based on the observable σ_x, while faulty measurements receive a uniform weight. The weights are normalized to form a probability distribution, enabling the particle filter to prioritize particles consistent with the observed data, effectively managing the mixture of Gaussian noise and random outliers.

Algorithm 4 (Importance Sampling Bootstrap)**.** This algorithm updates the particle set and their weights based on the measurement model for the spin observable.

- **Input:**
  - Previous particle set SX, number of particles N, evolution operator $U_{\Delta t}$, measurement parameters $\{Y, c_p, ub, lb, R\}$, time step k.

- **Propagate particles:**
  - For $i = 1$ to N:
    * Update particle: $|s_i\rangle \leftarrow U_{\Delta t}|s_i\rangle$.
  - Store updated particles in SX.

- **Compute weights:**
  - Sample indicators for measurement validity using Roulette Wheel (Algorithm 6) with probabilities $[c_p, 1 - c_p]$.
  - Identify indices ind_{correct} (valid measurements) and ind_{error} (faulty measurements).
  - For each $i \in ind_{\text{correct}}$:
    * Compute measurement: $y_{\text{valid},i} = \langle s_i|\sigma_x|s_i\rangle$.
    * Assign weight: $W_i = \frac{1}{\sqrt{2\pi R}} \exp\left(-\frac{(Y(k) - y_{\text{valid},i})^2}{2R}\right)$.
  - For each $i \in ind_{\text{error}}$:
    * Assign weight: $W_i = \frac{1}{ub - lb}$.
  - Normalize weights: $W_i \leftarrow \frac{W_i}{\sum_{j=1}^{N} W_j}$.

- **Output:**
  - Return updated particle set SX and weights W.

Resampling Algorithm: The Resampling algorithm mitigates particle degeneracy in the particle filter by selecting a new set of particles based on their weights. It normalizes the weights to sum to one and uses the Roulette Wheel selection method to sample N particles with replacement, favoring particles with higher weights. This process discards low-weight particles and replicates high-weight ones, maintaining a diverse and representative particle set. The weights are reset to $1/N$ to prevent bias in subsequent iterations. This resampling step is essential for sustaining the particle filter's accuracy, as it concentrates computational effort on particles that best represent the posterior distribution of the quantum state, improving the estimation of the spin observable σ_x.

Algorithm 5 (Resampling)**.** This algorithm resamples particles to mitigate degeneracy in the particle filter.

- **Input:**
  - Weights W, particle set SX.
- **Normalize weights:**
  - Compute $W_i \leftarrow \frac{W_i}{\sum_{j=1}^{N} W_j}$ for $i = 1$ to N.
- **Resample particles:**
  - Sample N indices Ind using Roulette Wheel (Algorithm 6) with probabilities W.
  - Construct new particle set: $new\_Sample\_set = SX(:, Ind)$.
- **Reset weights:**
  - Set $new\_weight = \frac{1}{N} \cdot [1, \ldots, 1]_{1\times N}$.
- **Output:**
  - Return new_Sample_set and new_weight.

Roulette Wheel Selection Algorithm: The Roulette Wheel Selection algorithm generates N samples from a discrete probability distribution, used in both importance sampling and resampling steps of the particle filter. It computes the cumulative distribution of the input probabilities (particle weights or measurement validity probabilities) and samples indices by generating uniform random numbers in $[0, 1)$ and selecting the smallest index where the cumulative probability exceeds the random value. This method ensures that indices are chosen proportional to their probabilities, simulating a "roulette wheel" where higher-probability outcomes are more likely to be selected. In the particle filter, this algorithm supports importance sampling by determining whether measurements are valid or faulty and resampling by selecting particles based on their weights, facilitating efficient handling of the probabilistic measurement model for the spin observable estimation.

Algorithm 6 (Roulette Wheel Selection)**.** This algorithm samples indices from a discrete probability distribution using the roulette wheel method.

- **Input:**
  - Probability distribution $Prob$, number of samples N.
- **Compute cumulative distribution:**
  - Calculate $A = \text{cumsum}(Prob)$.
- **Sample indices:**
  - Initialize output array $sample$ of size $1 \times N$.
  - For $i = 1$ to N:
    - Generate uniform random number $x \sim U[0, 1)$.
    - Set $sample(i) = \min\{j \mid A_j > x\}$.
- **Output:**
  - Return sampled indices $sample$.

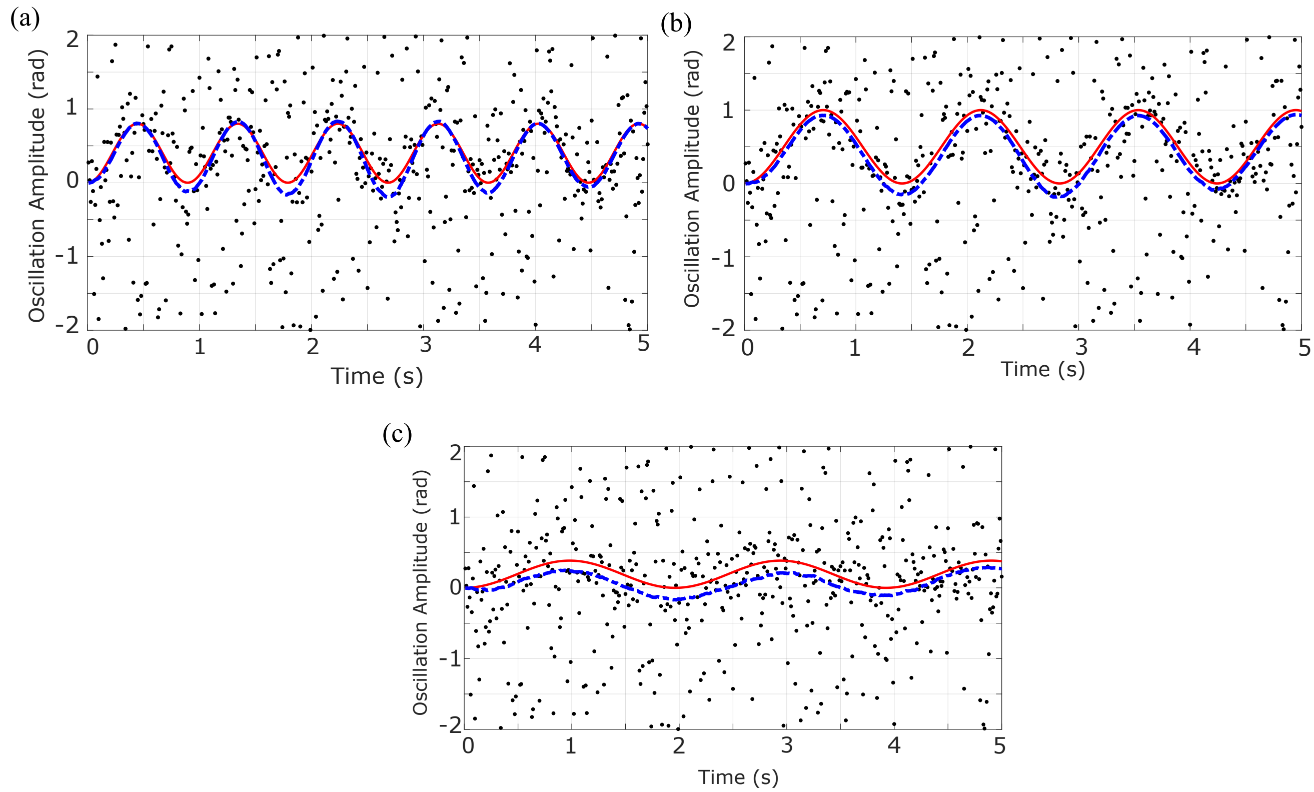

FIG. 1: Estimation of Pauli-x observable over 5 seconds using particle filtering algorithm. In all the figures, the black dotted points represent measurement data, the solid red line represents the theoretical prediction, and the dashed blue line represents the estimation by the particle filter. In Fig.(a), $\omega_f = 0$, i.e., $\Delta\omega = \omega_0$. The corresponding RMSE is 0.08. In Fig.(b), $\Delta\omega = 0.5\omega_0$. The corresponding RMSE is 0.10, and finally in Fig.(c), $\Delta\omega = 0.1\omega_0$. The corresponding RMSE is 0.14.

B. Results

We present our results in Fig. 1, where we estimate the oscillating Pauli-x observable using particle filter over 5 seconds. We show three cases of detuning: Fig.(a) for $\Delta\omega = \omega_0$, where we obtain RMSE = 0.08, in Fig.(b) we have $\Delta\omega = 0.5\omega_0$ and RMSE is 0.10, and in Fig.(c) $\Delta\omega = 0.1\omega_0$. The corresponding RMSE is 0.14. The detailed description of the figure is in the figure caption. In the next section, we present all the algorithms.

V. CONCLUSION

This work demonstrates the successful application of the Bootstrap Particle Filter to estimate the spin observable $\langle\sigma_x\rangle_{\tilde{\psi}(t)}$ of a two-level quantum system undergoing Rabi oscillations, despite challenging measurement noise comprising both Gaussian fluctuations (variance $R = 0.1$) and random outliers (uniformly distributed in $[-2, 2]$ with probability 0.5). By mapping the quantum state evolution, governed by the time-independent Hamiltonian $\tilde{H}$ in the rotating frame, to a classical state-space model with deterministic dynamics, we effectively adapted the particle filter algorithm to track the oscillatory behaviour of $\langle\sigma_x\rangle_{\tilde{\psi}(t)}$. The algorithm, utilizing $N = 10{,}000$ particles, achieved robust estimation across different detuning parameters ($\Delta\omega = \omega_0, 0.5\omega_0, 0.1\omega_0$), with root mean square errors (RMSE) of 0.08, 0.10, and 0.14, respectively, as shown in Fig. 1. These results highlight the algorithm's ability to handle Gaussian noise and outliers, a common challenge in quantum experiments.

The primary contribution of this work lies in extending classical particle filtering to quantum state estimation, leveraging the deterministic evolution of the spin observable under Schrödinger's equation. The observed increase in RMSE with decreasing $\Delta\omega$ suggests that smaller detunings, which reduce the oscillation amplitude, amplify the impact of measurement noise, a finding that warrants further investigation.

Limitations include the assumption of deterministic dynamics, which may not fully capture real-world scenarios where the Hamiltonian is subject to stochastic perturbations (e.g., magnetic field fluctuations). Additionally, the computational cost of using $N = 10{,}000$ particles may be prohibitive for real-time applications. Future research could explore:

- Extending the model to incorporate stochastic dynamics, such as those described by the stochastic Schrödinger equation, to account for environmental noise.
- Optimizing the number of particles or exploring advanced resampling techniques (e.g., systematic resampling) to reduce computational complexity while maintaining accuracy.
- Investigating adaptive measurement strategies to mitigate the impact of faulty measurements, potentially reducing the RMSE for small $\Delta\omega$.
- Applying the particle filter to multi-qubit systems or other quantum observables, broadening its applicability in quantum information processing.

In conclusion, this work establishes the particle filtering algorithm as a powerful tool for quantum state estimation under realistic noise conditions, offering a foundation for further advancements in quantum control and measurement.

[1] S. Särkkä, *Bayesian Filtering and Smoothing* (Cambridge University Press, 2013).
[2] J. J. Sakurai and J. Napolitano, *Modern quantum mechanics*, 3rd ed. (Cambridge University Press, Cambridge, 2021).