We thank reviewers for the constructive comments. Reviewers found our paper "(very) well written" (R1-R4), "a significant contribution" (R2), "a worthwhile endeavour" (R1), and "quite sound" (R3). Please find our response below.

[R1] Probability of collision for a full trajectory: Given n_t as the total number of time steps, the probability of collision avoidance between robot i, j for the whole trajectory is lower bounded as $\Pr\big(\bigcap_{t=1}^{n_t}(\mathbf{x}_i^t, \mathbf{x}_j^t \in \mathcal{H}_{i,j}^s(t))\big) = \prod_{t=1}^{n_t}\Pr(\mathbf{x}_i^t, \mathbf{x}_j^t \in \mathcal{H}_{i,j}^s(t)) \geq \sigma^{n_t}$. In theory, by selecting $\sigma = \exp(\frac{\ln\sigma_{all}}{n_t})$ one could achieve a lower bounded joint collision free threshold of σ_{all} for the full trajectory. However, it could be over conservative in the long run, e.g. step-wise threshold $\sigma = 0.9949$ leads to $\sigma_{all} = 0.6$ for $n_t = 100$. Hence step-wise threshold is used more often to construct local collision constraints (see [8,9,13,29,30,35,36]). An alternative is to impose discounting factor $\beta < 1$ so that the penalty of future violation probabilities is relaxed, i.e. step-wise threshold σ renders the same bounded joint threshold for the whole trajectory $\sum_{t=1}^{n_t}(\beta)^t\Pr(\mathbf{x}_i^t, \mathbf{x}_j^t \in \mathcal{H}_{i,j}^s(t)) \geq \sigma$ if given discounting factor $\beta > 0.5$ (see [36]). We will provide new results in the updated version with different conservativeness levels of σ, σ_o to give more insights.

[R1] Relevance to NeurIPS community: Besides control community, we believe the work is well connected to NeurIPS community as well. As the real world is inherently stochastic, it is important to derive mathematically correct safety consideration accounting for uncertainties. In particular, we think the presented work could bridge learning based methodologies and model based safety-critical control with formally provable safety guarantee. For example, a very recent work [37] presents an episodic learning framework to model the partially known dynamics and uses control barrier functions similar to the *deterministic* version of our approach for safety consideration *without* noisy uncertainties. Another example: one may use learning techniques such as Gaussian Processes to learn a partially unknown dynamical system with noisy uncertainties and use our approach to compute a certified probablistically safe policy to collect more data for further improving models. We think integrating dynamical system learning with our PrSBC framework to guarantee safe learning to control is an important future direction.

[R2] Generality of the method: Our model is general and can be applied to stochastic dynamical systems in control affine form (1) which captures a large family of dynamical systems, such as 3-dof unicycle dynamics [19,28], 12-dof quadrotors [29,33] (both already evaluated in our experiments), bipedal robots, automotive vehicle, and Segway robots [4,37]. For heavy unicycle with low ground friction, our method is applicable as the robot could be described by a nominal unicycle dynamics with limited acceleration [28] due to inertia and low friction with noises.

[R2] Step-wise QP optimization: The reviewer is correct about our step-wise optimization process (15). Our safe controller $\mathbf{u}(\mathbf{x})$ is dependent on each visited robot state $\mathbf{x}$ and if the safe control $\mathbf{u}(\mathbf{x}(t)) \in \mathcal{S}_{\mathbf{u}}^{\sigma} \bigcap \mathcal{S}_{\mathbf{u}}^{\sigma_o}$ for all $t \in [0, \tau]$, then it has chance constrained guarantee for any $\mathbf{x}(t) \in \mathcal{H}^s$ within $t \in [0, \tau]$, which folllows the definition of the forward-invariance (see our Lemma 1 and Theorem 2 in [4]). From a continuous-time perspective, QP with PrSBC constraints in (15) per time step ensures for all $t \in [0, \tau]$, $\mathbf{u}(\mathbf{x}(t)) \in \mathcal{S}_{\mathbf{u}}^{\sigma} \bigcap \mathcal{S}_{\mathbf{u}}^{\sigma_o}$, then our approach guarantees chance constrained safety along the entire time horizon $[0, \tau]$, not just a particular time point.

[R3] Expressivity of the framework: While we address safety on collision avoidance by enforcing minimum distance to define the control barrier function h^s (2), the existence of PrSBC constraints in our Theorem 3 are general and do not rely on the form of h (see proof and (11)). For other safety consideration, we can apply our algorithm to different forms of task-specific control barrier function h as in Section V in [4] (see V.D for battery charging constraint example). Recent work [38] has employed the deterministic control barrier functions to signal temporal logic (STL) formulations. Our approach could be used in the same way with STL for explicit temporal safety considerations with uncertainty.

[R3] Comparison to distributed robot control: As discussed in [4], the traditional Lyapunov approach (as in the referred book) handles stability that drives a dynamical system to a point or a overly restrictive sublevel set describing *final* spatial configurations. However, safety is often framed as *enforcing invariance* of a permissive set, i.e. starting from and not leaving a safe set. Furthermore, our contribution is the novel PrSBC framework that extends deterministic safety barrier certificate approach to a probabilistic setting with formally proved bounded safety guarantee. This is significantly different from the work in the book that does not address uncertainties, safety, nor collision avoidance.

[R4] Probabilistic model used for evaluation: In evaluation we use uniform distributions with finite support as the probabilistic model and randomly sample from the distributions per time step to simulate the stochastic dynamics and observations. The control space is $\mathbb{R}^m$ with $m \leq d$. In the example 2, the larger error box for robot 1 indicates a greater uniform error with larger bounded support, which is randomly selected for testing purposes.

[37] Taylor, Andrew, Andrew Singletary, Yisong Yue, and Aaron Ames. "Learning for safety-critical control with control barrier functions." In *Learning for Dynamics and Control*, pp. 708-717. 2020.

[38] Lindemann, Lars, and Dimos V. Dimarogonas. "Control barrier functions for signal temporal logic tasks." *IEEE control systems letters* 3.1 (2018): 96-101.