**We thank the reviewers for their comments and insightful reviews.** We will integrate all the useful suggestions in the revised version of the paper.

**[R1] Comparison with reward-free exploration [23].** On a high level, both approaches build accurate estimates of the transitions on a specific (unknown) state space of interest: "significant" states within $H$ steps for [23] and incrementally $L$-controllable states $S_L^{\rightarrow}$ for DISCO. While the two concepts are somewhat related ($H \equiv L$ are the horizons of interest), [23] focuses on finite-horizon problems and we consider the more general goal-conditioned setting. Resetting after every $L$ steps (as in finite-horizon) would not allow identifying the states in $S_L^{\rightarrow}$. This explains the distinct technical tools used: while [23] deploys finite-horizon no-regret algorithms, DISCO leverages SSP tools. The bound-wise comparison is also interesting. While $\epsilon$, $A$ and $H \equiv L$ dependencies match, [23]'s dependency on the global state space $S$ is polynomial, whereas DISCO's is only *logarithmic* as the main dependency is w.r.t. $|S_{L+\epsilon}^{\rightarrow}|$. This shows that DISCO effectively adapts to the state space of interest and it ignores all other states.

**[R1] Computational complexity.** The overall complexity can be expressed as $\sum_{k=1}^{K} |\mathcal{W}_k| \cdot C(\text{OVI}_{\text{SSP}})$, with $C(\text{OVI}_{\text{SSP}})$ the complexity of an $\text{OVI}_{\text{SSP}}$ procedure. Note that $K \leq |S_{L+\epsilon}^{\rightarrow}|$ and $|W_k| \leq 2LA|\mathcal{K}_k| \leq 2LA|S_{L+\epsilon}^{\rightarrow}|$. The VI algorithm for SSP was proved in [37] to converge in time quadratic w.r.t. the size of the considered state space (here, $\mathcal{K}_k$) and $\|V^\star\|_\infty / c_{\min}$. Here $c_{\min} = 1$, and we can prove that in all SSPs considered by DISCO, the optimal value function $V^\star$ verifies $\|V^\star\|_\infty = O(L^2)$ due to the restriction of the goal in $\mathcal{W}_k$. Putting everything together gives DISCO's complexity. Interestingly, it only depends on $|S_{L+\epsilon}^{\rightarrow}|$ and is independent from the global state space size $S$.

**[R1, R3] Motivation/limitations of the incremental framework.** We believe this setting effectively captures the intuition that an agent progressively expands its knowledge of the environment by leveraging closer well-controlled states to achieve further states that are more difficult to reach. Interestingly, recent goal-conditioned algorithms for unsupervised RL or learning with sparse reward (see e.g., [21,22,33]) make the *implicit* assumption that the environment's states satisfy the incremental controllability condition of Def. 4, in the sense that they strive to train a policy to reach closer states before moving forward in exploring and controlling other states. Nonetheless, the definition of $S_L^{\rightarrow}$ may be too restrictive as it excludes states that are $L$-controllable but may require passing through states that are not. While considering all $L$-controllable states in $\mathcal{S}_L$ would inevitably hit the impossibility result proved by [1], we believe it is possible to relax the strict incrementality condition of $S_L^{\rightarrow}$ without affecting the learnability of the problem.

**[R1] On $L$.** In DISCO we can gradually increase the value of $L$ without restarting the algorithm from scratch, unlike in UcbExplore. This allows tuning the parameter online according to the desired behavior. In particular, in the case of communicating MDPs, one may perform a sort of doubling trick: $L = 2, 4, 8, \ldots, 2^n$, where the unknown $n$ satisfies $2^{n-1} \leq D \leq 2^n$. Once $2^n$ the algorithm would indeed discover all states in the MDP and we can stop it. Crucially, the total sample complexity would be (up to logarithmic factors) the same of DISCO run with the final value of $L$.

**[R1] Upper limit on $\epsilon$.** We had set $\epsilon \leq 1$ for ease of analysis, as assumed in e.g., [35]. If it may larger ($\epsilon \leq \epsilon_{\max}$), then the definition of $\mathcal{W}_k$ would indeed have to be modified accordingly (replacing $1 - \epsilon/2$ by $1 - \epsilon/(2\epsilon_{\max})$).

**[R2] Proof sketch.** A sketch of the proof of Thm. 1 is currently available in App. B. In case of acceptance we will use the extra page to bring it to the main text so that it indeed contains proof intuition (likewise for Cor. 1).

**[R2] Additional experiments.** Since [1] did not report any numerical study of UcbExplore, in our paper we focused on two simple environments where it is still relatively easy to interpret the behavior of the algorithms and their performance. We will include additional experiments for varying $L$ in the final version.

**[R3] Bound dependencies, comparison.** In the condition on line 224, the number of states $S_{L+\epsilon} := |S_{L+\epsilon}^{\rightarrow}|$ directly depends on $L$ and $\epsilon$ (more precisely it increases with both). As such, all parameters are connected to each other and it may not be trivial to determine values for which the condition holds. In the environments considered in our experiments, the condition holds for the chosen values of $\epsilon$ and $L$. We agree an interesting direction for future investigation is to identify *families* of MDPs where $S_{L+\epsilon}$ is an explicit function of $L$ (e.g., constant, linear, polynomial, exponential).

**[R3] DISCO for cost-sensitive tasks.** We agree the current discussion is poorly phrased. DISCO indeed does not perform any additional learning. In fact, DISCO returns an estimated model (on which OVI is run) that is sufficiently accurate w.r.t. the true model restricted to $S_L^{\rightarrow}$. This property guarantees that the SSP policy returned by OVI is near-optimal for any cost function. Interestingly, it may also be used to compute accurate policies for e.g., finite-horizon RL tasks restricted on $S_L^{\rightarrow}$ (by leveraging the simulation lemma of Lem. 8). We will clarify this part.

**[R4] Complete bound.** We can retrace the exact terms from the analysis to provide a sample complexity bound with constants and logs. This will indeed allow to evaluate the bound w.r.t. the performance. We note that while the performance is partially tied to the bound via the choice of allocation $\phi$, samples are in practice shared between sample-collection attempts, and in addition the $O(L)$ cost to collect each sample is often loose for states close to $s_0$.

**[R4]** "$L$-reachability" and "$L$-controllability" are indeed the same concept, we will unify the terminology.