type of decoupling technique, creating single agent problems that allow for much better scaling up. We also see here that domain specific techniques can help POMDP solvers in creating better policies, with respect to the synchronization of policies. Our localization additions were successful because, when considering a single agent, localization is less important, unless we are near the goal. However, when other agents are nearby, localization can reduce significantly the possibility for collisions, and hence, the need to replan, and the constraints during replanning.

6 Related Work

MAPF with unexpected delays has been studied in different forms (**????**). Yet limiting stochastic effects to delays simplifies the problem compared to SMAPF-PO, and these works all assumed perfect observability. UM^* (**?**) does support uncertainty over agents' locations but it allows agents to risk conflicts as long as their probability of occurring is beneath a given threshold. Similarly, prior work on multi-robot navigation assigned cost to collision and aim to minimize it. The SMAPF-PO is fundamentally different since we require avoiding any chance for having a conflict. Recent work (**?**) on MAPF with partial observability differs from SMAPF-PO in that they assumed control is not centralized and agents' visibility is based on line of sight. The latter assumption is different from our beacons-based observation model, precluding empirical comparison. MAPF under movement uncertainty is a special case of Multi-Agent MDP (MMDP) (**?**). General-purpose MMDP solvers (**?**) do not exploit the specific structure of the SMAPF-PO problem, such as the limited interactions between agents. Similarly, online approaches for solving large MDPs, such as FF-Replan(**?**), are not expected to be effective in our domain and do not deal with partial observability. Dec-POMDP (**?**) is often used to model multi-agent problems under partial observability, where agents plan jointly, but execute their policies independently. This is different from our setting, where agents are controlled in a centralized manner and share their observations. Also, Dec-POMDP algorithms tend to scale poorly unless the interactions between the agents are limited to small predefined areas in the state space (**?**). Multiagent POMDP (**?**), where both planning and control are centralized, is an appropriate model for SMAPF-PO. **?** suggest an algorithm based on POMCP which exploits the locality of agent interactions. Their algorithm given singleton sets is similar to our POMCP, which does not scale as well as FSVI. Our approach can be viewed as a particular implementation of FT-POMCP leveraging the structure of the SMAPF-PO problem for factorization, and using prioritized planning to ensure independence. Factorizing the decision-making process in POMDPs has long roots (**??**, e.g.), and is particularly useful given several independent tasks. Factorization is less effective, however, when the coupling between the tasks increases, as happens in our problem where agents may collide. In tasks with high uncertainty concerning the current state, such coupling is common, and useful factoring is difficult. We used FSVI and POMCP but other alternatives for offline and online POMDP solvers exist (**???**, e.g.). Our contribution is not in an adaptation of a particular POMDP solver to SMAPF-PO, but in our factoring and prioritization schemes. Replacing POMCP with, e.g., ABT, is likely to scale up only slightly, while improvements to the factorization can make a huge difference.

P	Alg	ADR	RT	% S	#C	#R
M_1	FSVI	69 ± 4	21	74%	0.3±0.1	0.4±0.2
	POMCP+FL	52 ± 5	33	62%	15±12	16±13
	FSVI+MF	70 ± 4	19	78%	0.4±0.2	0.6±0.2
	FSVI+FL	61 ± 5	22	66%	0.2±0.1	0.3±0.2
	FSVI+MF+FL	70 ± 4	18	78%	0.4±0.2	0.6±0.2
M_2	FSVI	41 ± 5	146	46%	4±0.4	7±0.6
	POMCP+FL	28 ± 5	946	46%	300±20	300±20
	FSVI+MF	28 ± 4	230	30%	5±0.3	7±0.4
	FSVI+FL	48 ± 5	191	64%	6±0.5	9±0.6
	FSVI+MF+FL	55 ± 4	200	76%	5±0.4	8±0.5
M_3	FSVI	17.8 ± 0.4	253	0%	1±0	2±0
	POMCP+FL	57 ± 3	202	84%	79±15	83±15
	FSVI+MF	17.6 ± 0.3	418	0%	1±0	2±0
	FSVI+FL	56 ± 3	402	92%	10±1	16±3
	FSVI+MF+FL	48 ± 4	535	72%	10±2	18±3
M_4	FSVI	30 ± 3	133	10%	1±0	2±0
	POMCP+FL	73 ± 2	208	94%	63±16	63±16
	FSVI+MF	36 ± 4	91	20%	1±0	2±0
	FSVI+FL	77 ± 7	245	96 %	2±0.3	4±0.4
	FSVI+MF+FL	75 ± 3	125	94%	3±0.7	5±1
M_5	FSVI	-6 ± 3	381	0%	2.1±0.2	3.4±0.2
	POMCP+FL	-11.6 ± 0.6	275	2%	55±5	101±3
	FSVI+MF	5 ± 2	217	0%	2.0±0.1	3.3±0.2
	FSVI+FL	-6 ± 2	896	8%	29±3	55±6
	FSVI+MF+FL	0 ± 2	671	2%	20±3	37±5
L_1	FSVI	37 ± 6	254	8%	7.2±0.5	10.9±0.8
	POMCP+FL	64 ± 5	1954	64%	372±21	376±21
	FSVI+MF	29 ± 6	433	6%	6.8±0.5	10.5±0.6
	FSVI+FL	97 ± 3	222	78%	8±0.5	12±0.7
	FSVI+MF+FL	97 ± 4	386	74%	10±3	15±4
L_2	FSVI	-11 ± 2	80	0%	2.96±0.02	8.8±0.1
	POMCP+FL	69 ± 12	1950	26%	-	-
	FSVI+MF	-15.1 ± 0.3	57	0%	3±0	8.9±0.1
	FSVI+FL	67 ± 12	182	36%	25±3	50±6
	FSVI+MF+FL	73 ± 11	212	30%	20±3	40±6
L_3	FSVI	67 ± 6	202	8%	8±0.6	12±0.8
	POMCP+FL	35 ± 7	1663	12%	328±23	326±22
	FSVI+MF	59 ± 7	353	8%	9±1	13±1
	FSVI+FL	112 ± 5	183	62%	10±1	14±1
	FSVI+MF+FL	117 ± 6	423	68%	12±1	18±2

Table 3: Results for medium and large problems.

7 Conclusion

We studied the Stochastic MAPF with Partial Observability (SMAPF-PO) problem, which is a generalization of MAPF in which actions have stochastic outcomes and agents do not have perfect observability of the current state. We focused on a centralized control setup. While SMAPF-PO can be modeled as a single-agent POMDP problem, solving this POMDP is intractable even for small problems. We introduced the OPP approach, an online adaptation of Prioritized Planning. OPP has several non-trivial components which we describe, and we also propose two extensions that encourage the agents to actively localize. The results showed that OPP solves larger problems than an offline baseline. Yet, solving larger problems is still an open challenge for future work. Possimus placeat accusamus corporis officia temporibus, harum consequatur quidem.A pariatur quo maxime blanditiis porro dolores quos, eligendi odit nobis veritatis cumque vitae accusamus molestias eos est blanditiis, molestiae commodi totam temporibus optio perspiciatis itaque tenetur cupiditate?Consequuntur voluptas tempora voluptatem, nisi aliquam minus deserunt deleniti alias illo minima quis, assumenda vero perferendis aperiam culpa amet quasi dolorum officia alias?Nemo porro saepe dolores rerum exercitationem, quo magnam amet totam, nisi enim vitae aliquid officia iste tempore explicabo, eaque deleniti iusto odit similique dolor, ex dicta exercitationem adipisci distinctio alias deserunt qui dolore aut.