was less than 15 ($\approx$ solving SOLVEDSPAIT-LINEAR() 15 times). Since SOLVEDSPAIT-PWLC does not provide optimal solutions, the key results are with respect to solution quality. While, SOLVEDSPAIT-PWLC converges to local optima , a very important and practically interesting phenomenon is observed with respect to the quality of the local optima. Figure 5 provides results on all our experiments with PWLC reward functions with number of components ranging from 5 - 20. Here are the key results:
(a) The total number of iterations for convergence (the number of times the while loop in SOLVEDSPAIT-PWLC() is executed) varied between 9-14. However, the number of iterations required to be near local optima was only 3 or 4.
(b) For each problem , we started with 10 random starting values of **k** and as witnessed in all the three graphs, the starting solution quality has a very high variance. However, the algorithm converged to local optima that were very close to each other on all the problems (3 shown here and numerous others that we experimented with). While we do not know if the global optima is close to these set of local optima, this is a unique result for a local optimal algorithm, especially since the problems were generated randomly.

Previously, SMFU was proposed by (Varakantham et al. 2012) to compute equilibrium solutions for stochastic decision making problems for competitive settings. SMFU exploits the anonymity in interactions while computing equilibrium and hence our comparison against SMFU. We compare against SMFU and not against D-TREMOR (Velagapudi et al. 2011) because:

- SMFU employs shaping of model based on influences of other agents, similar to D-TREMOR.
- SMFU exploits anonymity in interactions and scales to problems with thousands of agents.
- Unlike D-TREMOR, SMFU converges to local optimal.

Since SMFU's solution depends on the initial policy, performance was averaged over multiple initializations of the starting policy. The SOLVEDSPAIT-LINEAR() and SOLVEDSPAIT-PWLC() computed optimal policies at run-times (at least) an order of magnitude faster than the runtime by SMFU. For instance, on the 80x80x24 problem (equivalent in size to the real world taxi fleet optimization problem in (Varakantham et al. 2012)), the SOLVEDSPAIT-LINEAR problem generated optimal solutions in 70 seconds, whereas SMFU took close to 30 minutes. However, with respect to SOLVEDSPAIT-PWC(), SMFU computed solutions in runtimes that were an order of magnitude shorter on large problems (ex: 30x10x10). While SMFU returned optimal solutions in few cases, overall it returned solutions that were around 60% of the optimal.

Acknowledgements

This research is supported in part by the National Research Foundation (NRF) Singapore through the Singapore MIT Alliance for Research and Technology (SMART) and its Future Urban Mobility (FM) Interdisciplinary Research Group.

References

Ahmed, A.; Varakantham, P.; Adulyasak, Y.; and Jaillet, P. 2013. Regret based robust solutions for uncertain markov decision processes. In *NIPS'13*.

Ahmed, A.; Varakantham, P.; and Cheng, S.-F. 2012. Decision support for agent populations in uncertain and congested environments. In *UAI'12*, 44–53.

Becker, R.; Zilberstein, S.; Lesser, V.; and Goldman, C. 2004. Solving transition independent decentralized Markov decision processes. *Journal of Artificial Intelligence Research* 22:423–455.

Bernstein, D.; Givan, R.; Immerman, N.; and Zilberstein, S. 2002. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research* 27(4):819–840.

Bernstein, D. S.; Hansen, E. A.; and Zilberstein, S. 2005. Bounded policy iteration for decentralized POMDPs. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence*, 1287–1292.

Hooker, J. 1995. Logic-based benders decomposition. *Mathematical Programming* 96:2003.

Kumar, A.; Zilberstein, S.; and Toussaint, M. 2011. Scalable multiagent planning using probabilistic inference. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, 2140–2146.

Melo, F. S., and Veloso, M. M. 2011. Decentralized mdps with sparse interactions. *Artif. Intell.* 175(11):1757–1789.

Mostafa, H., and Lesser, V. 2012. Offline planning for communication by exploiting structured interactions in decentralized mdps. In *IAT'09*.

Nair, R.; Varakantham, P.; Tambe, M.; and Yokoo, M. 2005. Networked distributed POMDPs: A synthesis of distributed constraint optimization and POMDPs. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 133–139.

Roughgarden, T., and Tardos, É. 2002. How bad is selfish routing? *Journal of the ACM* 49(2):236–259.

Seuken, S., and Zilberstein, S. 2007. Improved memory-bounded dynamic programming for decentralized POMDPs. In *UAI*.

Shieh, E.; Jain, M.; Jiang, A. X.; and Tambe, M. 2013. Ef?ciently solving joint activity based security games. In *International Joint Conference on Artificial Intelligence (IJCAI)*.

Varakantham, P.; Kwak, J. Y.; Taylor, M.; Marecki, J.; Scerri, P.; and Tambe, M. 2009. Exploiting coordination locales in distributed POMDPs via social model shaping. In *Nineteenth International Conference on Automated Planning and Scheduling*.

Varakantham, P.; Cheng, S.-F.; Gordon, G.; and Ahmed, A. 2012. Decision support for agent populations in uncertain and congested environments. In *AAAI*, 1471–1477.

Velagapudi, P.; Varakantham, P.; Sycara, K.; and Scerri, P. 2011. Distributed model shaping for scaling to decentralized POMDPs with hundreds of agents. In *AAMAS*, 955–962.

Witwicki, S., and Durfee, E. 2012. Influence-based policy abstraction for weakly-coupled dec-pomdps. In *ICAPS'10*.

Yin, Z., and Tambe, M. 2011. Continuous time planning for multiagent teams with temporal constraints. In *International Joint Conference on Artificial Intelligence (IJCAI)*.