Anytime Multi-Agent Path Finding via Large Neighborhood Search: Extended Abstract
Jiaoyang Li (jiaoyanl@usc.edu), 1 Zhe Chen, 2 Daniel Harabor, 2 Peter J. Stuckey, 2 Sven Koenig 1
1 University of Southern California, 2 Monash University

Abstract
Multi-Agent Path Finding (MAPF) is the challenging problem of computing collision-free paths for multiple agents. MAPF algorithms can be categorized on a spectrum. At one end are (bounded-sub)optimal algorithms that can find high-quality solutions for small problems. At the other end are unbounded-suboptimal algorithms that can solve very large practical problems but usually find low-quality solutions. In this paper, we consider a third approach that combines both advantages: anytime algorithms that quickly find an initial solution, including for large problems, and that subsequently improve the solution to near-optimal as time progresses. To improve the solution, we replan subsets of agents using Large Neighborhood Search. Empirically, we compare our algorithm MAPF-LNS to the state-of-the-art anytime MAPF algorithm anytime BCBS and report significant gains in scalability, runtime to the first solution, and speed of improving solutions.

2 MAPF-LNS
Large Neighborhood Search (LNS)
LNS[2] combines the power of Constraint Programming (CP) (or Mixed Integer Programming) and Local Search (LS).

- Initialize: Find a final solution (CP) by any non-optimal MAPF solver.
- Destroy: Select a subset of agents $A_i$.
- Repair:
  - Fix the paths for the agents not in $A_i$ and plan collision-free paths for the agents in $A_i$ (by a modified MAPF solver).
  - Replace the old paths if the new ones result in a smaller sum of the travel times.

MAPF-LNS
MAPF-LNS is an anytime MAPF algorithm motivated by LNS.

- Initialize: A MAPF solution (by any non-optimal MAPF solver).
- Destroy: Select a subset of agents $A_i$.
- Repair:
  - Fix the paths for the agents not in $A_i$ and plan collision-free paths for the agents in $A_i$ (by a modified MAPF solver).
  - Replace the old paths if the new ones result in a smaller sum of the travel times.

Neighborhood: Fix a subset of variables to their values in the best solution found so far.

Adaptive LNS (ALNS)
ALNS[1] makes use of multiple destroy heuristics by recording their relative success in improving solutions and choosing the next neighborhood to explore guided by the most promising heuristic.

4 Empirical Evaluation

Warehouse
![Success rate vs. Number of agents]

Game
![Success rate vs. Number of agents]

Summary: On easy instances, that anytime BCBS can solve, MAPF-LNS has higher success rates, smaller runtimes to the first solution, and better final solutions than anytime BCBS. On hard instances, that anytime BCBS cannot solve, MAPF-LNS can rapidly improve the costly initial solution and quickly converge to a near-optimal solution.

Acknowledgement
Jiaoyang Li performed the research during her visit to Monash University. The research at the University of Southern California was supported by the National Science Foundation (NSF) under grant numbers 1409987, 1724392, 1817189, 1837779, and 1935712 as well as a gift from Amazon. The research at Monash University was supported by the Australian Research Council under Discovery Grant DP190100013 and DP200100025 as well as a gift from Amazon.