Table 3: **PF/TPF** with redundancy elimination (E) and circle heuristics (R) on random $25{\times}25$ grid graphs with $k=15$ agents and $o=20\%$ obstacles.

E	R	Grounding sec	First Solution sec (plan length)	Optimal Solution sec (plan length)
		3.48	2.44 (34.8)	19.32 (34.4)
X		16.71	5.23 (36.9)	56.24 (34.4)
	X	1.87	0.95 (37.1)	14.47 (34.5)†
X	X	17.18	2.11 (38.4)	30.33 (34.5)‡

† 2 instances unsatisfiable ‡ 2 instances out of memory

possible for two agents to move in opposite directions on an edge (v, u) in the given graph, but it may be possible in the environment for one agent to move from v to u and the other agent to move from u to v following different continuous trajectories. On the other hand, for some other applications where robots move via narrow roads, $\underline{\mathbf{X}}$ may be required.

We also experimented with other constraints: Adding $\underline{\mathbf{I}}$ to the ASP formulation of **PF** increases the computation time in many problems. Note that since $\underline{\mathbf{I}}$ ignores the time of an agent visiting a vertex, it may be too strong for many **PF** applications. Adding $\underline{\mathbf{L}}$ to the ASP formulation of **PF** similarly increases the computation time in many problems.

Experiments with Heuristics We utilized the "circle heuristic" (Erdem and Wong 2004) to improve the computational efficiency in terms of computation time and consumed memory. The circle heuristic identifies, for each agent, a subgraph of the given graph that is more "relevant" for that agent to search for a path: we introduce two "circles" with a given radius around the start and the goal positions of the agent, and require that the path connecting the start and the goal positions is contained in the union of these two circles. The radius can be defined as a constant, or a function of some distance between the start and the goal positions. By preprocessing, for each agent i, we identify the relevant edges (v, u) of the graph and represent them as facts of the form $\mathit{relevantEdge}(i, v, u)$; and replace in P all atoms of the form $\mathit{edge}(v, u)$ by $\mathit{relevantEdge}(i, v, u)$.

We also experimented with a "redundancy elimination" heuristic to eliminate certain redundant moves of agents; moving from vertex u to v via other vertices is not allowed if an edge (u, v) exists, except if the agent is waiting at u or v:

$$\begin{aligned} \leftarrow\ & \mathit{path}(i,t,u), \mathit{path}(i,t',v), \mathit{edge}(u,v), \\ & \mathit{not}\ \mathit{path}(i,t{+}1,u), \mathit{not}\ \mathit{path}(i,t'{-}1,v). \end{aligned}$$

where $0 \le t < t' < l_i$, $t{+}1 < t'$, $1 \le i \le k$, $v, u \in V$, and $v \neq u$. This heuristic intuitively removes redundancies in paths, and thus it is expected to improve the quality of solutions by making average plan lengths smaller. Note that there still may be redundancies if the specified maximum plan length is not small enough.

To analyze the effect of adding these heuristics, we considered randomly generated instances of **PF**, over 25×25 grid graphs with $o=20\%$ obstacles and $k=15$ agents. With the circle heuristic, for each agent i, we considered a radius of $\lceil \frac{ED_i}{2} \rceil + 3$ where ED_i is the Euclidean distance between the start and the goal locations of agent i.

Table 3 shows the results of our experiments. Each row in this table shows averages over the same set of instances used in our experiments with constraints (Table 2). Here E and R denote the redundancy elimination and the circle heuristics.

Table 4: **PF/TPF** with circle heuristics (R) on a road network.

k	Grounding sec	First Solution sec (plan length)	Optimal Solution sec (plan length)
5	1.02	0.16 (29.5)	4.97 (24.4)
10	1.39	0.59 (32.0)	11.65 (29.4)
15	3.30	1.14 (35.6)	18.16 (32.5)
20	4.42	1.68 (34.9)	22.35 (32.7)
25	7.60	4.13 (39.2)	37.73 (34.6)

Table 5: $\mathbf{PF}_{\underline{\mathbf{W}}}$**/TPF** with circle heuristics (R) on game map.

k	Grounding sec	First Solution sec (plan length)	Optimal Solution sec (plan length)
5	22.94	0.29 (34.8)	9.07 (29.0)
10	56.48	0.81 (34.7)	23.14 (33.0)
15	109.09	1.25 (45.7)	29.45 (45.7)
20	80.76	1.14 (35.3)	26.71 (35.3)
25	119.90	1.82 (40.3)	40.48 (40.3)

We observe that, since the redundancy elimination heuristic adds further constraints to the problem, the grounding time increases. These constraints, furthermore, do not constrain the search space enough to allow the solver engine to find solutions faster; therefore, the solution time also increases. Contrary to what we expected, solution quality also becomes worse with redundancy elimination. The additional constraints seem to be misleading for the solver, resulting in worse solution quality and worse efficiency.

With the circle heuristic, only some parts of the graph are considered while computing a solution; therefore, this heuristic significantly reduces both the grounding time and the time to find an optimal solution. Recall, however, that with a small value of the radius, the circle heuristic is neither sound nor complete: the optimal solution found for **PF** with the circle heuristic may not be an optimal solution for **PF**; also there may be instances of **PF** that have some solutions, but using the circle heuristic eliminates all solutions.

Experiments on a Real Road Network The results of our experiments with randomly generated instances of **PF** on the road network, using the circle heuristics, are shown in Table 4.

We observe that, with an increasing number of agents, grounding time does not increase as fast as the time to prove optimality. Finding an initial solution is fast in these instances, and plan length averages show that initial solutions are often already optimal.

The time to prove optimality is greater than the time for finding an initial solution; and quality of initial and optimal solutions are close to each other. Therefore, on a road network, it might be advantageous to try to find some solution (rather than an optimal one).

We also observed that optimal solution lengths are as same as the lengths of optimal solutions computed without using the circle heuristic.

Experiments on a Real Game Terrain The results of experiments with randomized instances of $\mathbf{PF}_{\underline{\mathbf{W}}}$ on a game terrain, using the circle heuristics, are shown in Table 5.