

Collective Adaptation through Concurrent Planning: the Case of Sustainable Urban Mobility

Antonio Bucchiarone¹, Daniel Furelos-Blanco², Anders Jonsson², Fahmida Khandokar³ and Monjur Mourshed³

¹Fondazione Bruno Kessler ²Universitat Pompeu Fabra ³Cardiff University

Abstract

We address the challenges that impede collective adaptation in smart mobility systems by proposing a notion of **ensembles**. Ensembles enable systems with collective adaptability to be built as emergent aggregations of autonomous and self-adaptive agents. Adaptation in these systems is triggered by a run-time occurrence, which is known as an **issue**.

The **novel aspect** of our approach is, it allows agents affected by an issue in the context of a **smart mobility scenario** to adapt collaboratively with minimal impact on their own preferences through an **issue resolution** process based on **concurrent planning** algorithms.

Roles and Ensembles

Our approach to collective adaptation involves the following concepts:

- An **ensemble** is an emergent aggregation of autonomous and self-adaptive agents.
- Each agent is defined by a set of **roles**.
- Collaboration involves taking **actions** and generating **issues** (e.g. blocked streets that force agents to take alternative routes).
- When an issue arises, a role handles the issue using one of its **solvers**.

Concurrent Planning

We adopt the formalism of **temporal planning** [1, 2] to generate concurrent solutions.

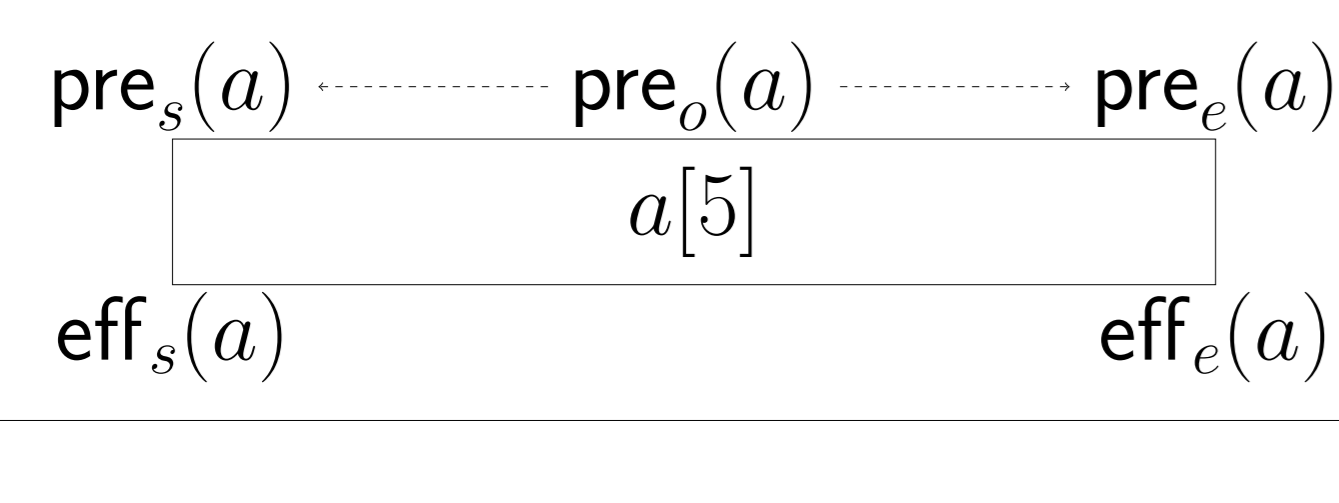


Figure: Model of an action in temporal planning.

Problem Modeling

There are two types of agents (passengers and carpools) distributed in a map. Each agent has an initial and a target location.

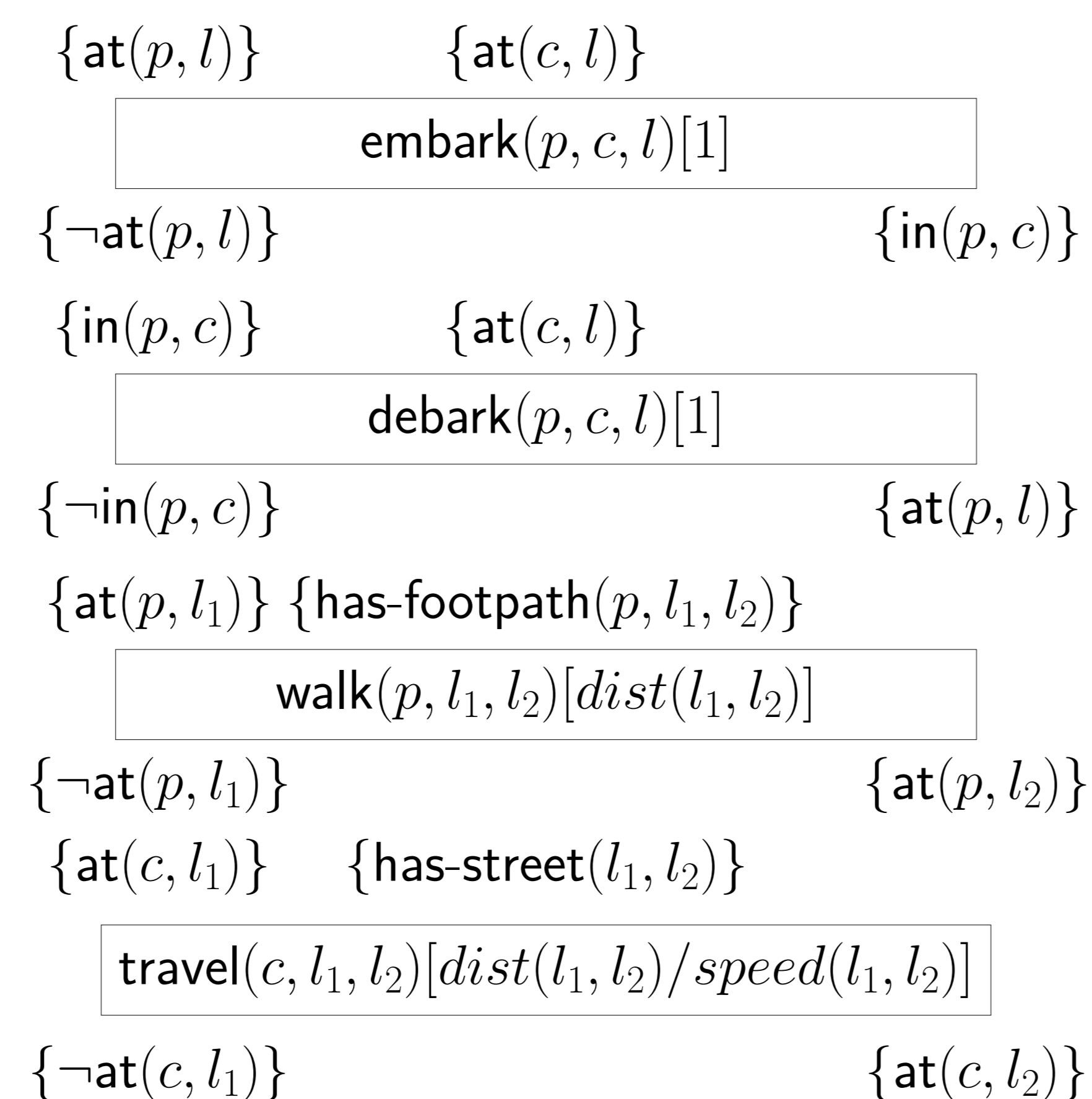


Figure: Temporal actions of the domain.

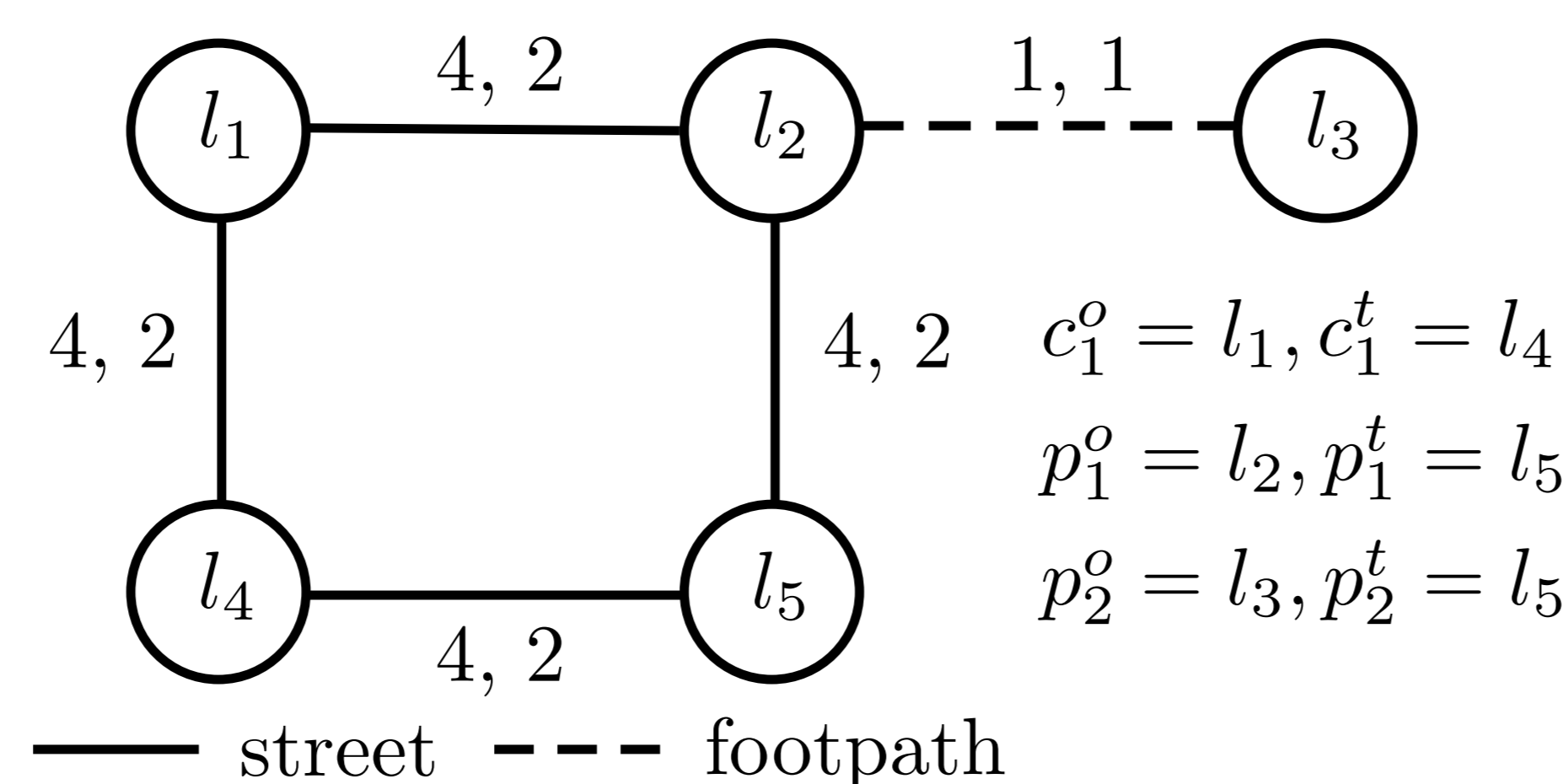


Figure: Simple smart mobility scenario.

start time	action	duration
0.0000	travel(c_1, l_1, l_2)	2.0000
0.0000	walk(p_2, l_3, l_2)	1.0000
2.0002	embark(p_1, c_1, l_2)	1.0000
2.0002	embark(p_2, c_1, l_2)	1.0000
3.0004	travel(c_1, l_2, l_5)	2.0000
5.0006	debark(p_2, c_1, l_5)	1.0000
5.0006	debark(p_1, c_1, l_5)	1.0000
6.0008	travel(c_1, l_5, l_4)	2.0000

Figure: Temporal plan for the previous scenario.

Evaluation

How are problems generated and solved?

- 1 Build problems using:
 - A real map of Trento obtained from OpenStreetMap [3].
 - A given number of agents (carpools and passengers).
- 2 Set random initial and target positions for the agents.
- 3 Convert the resulting scenarios into concurrent planning problems.
- 4 Solve the problems using the TPSHE planner [4].

How is evaluation done?

We generated 45 problems for different combinations of maps and number of agents:

- Maps of Trento with different number of links/streets: 2700, 5500 and 8200.
- The total number of agents ranged from 2 to 10.

The average solving time is measured for each combination of maps and number of agents.

Each experiment had a time limit of 5 minutes and a memory limit of 4 GB.

Results

↑ # agents, ↑ #links → ↓ # instances solved within budget.

- Small map (2700 links): 99.8%.
- Medium map (5500 links): 70.4%.
- Large map (8200 links): 39.6%.

Future work: Use a hierarchical approach to reduce the number of streets and decrease time.

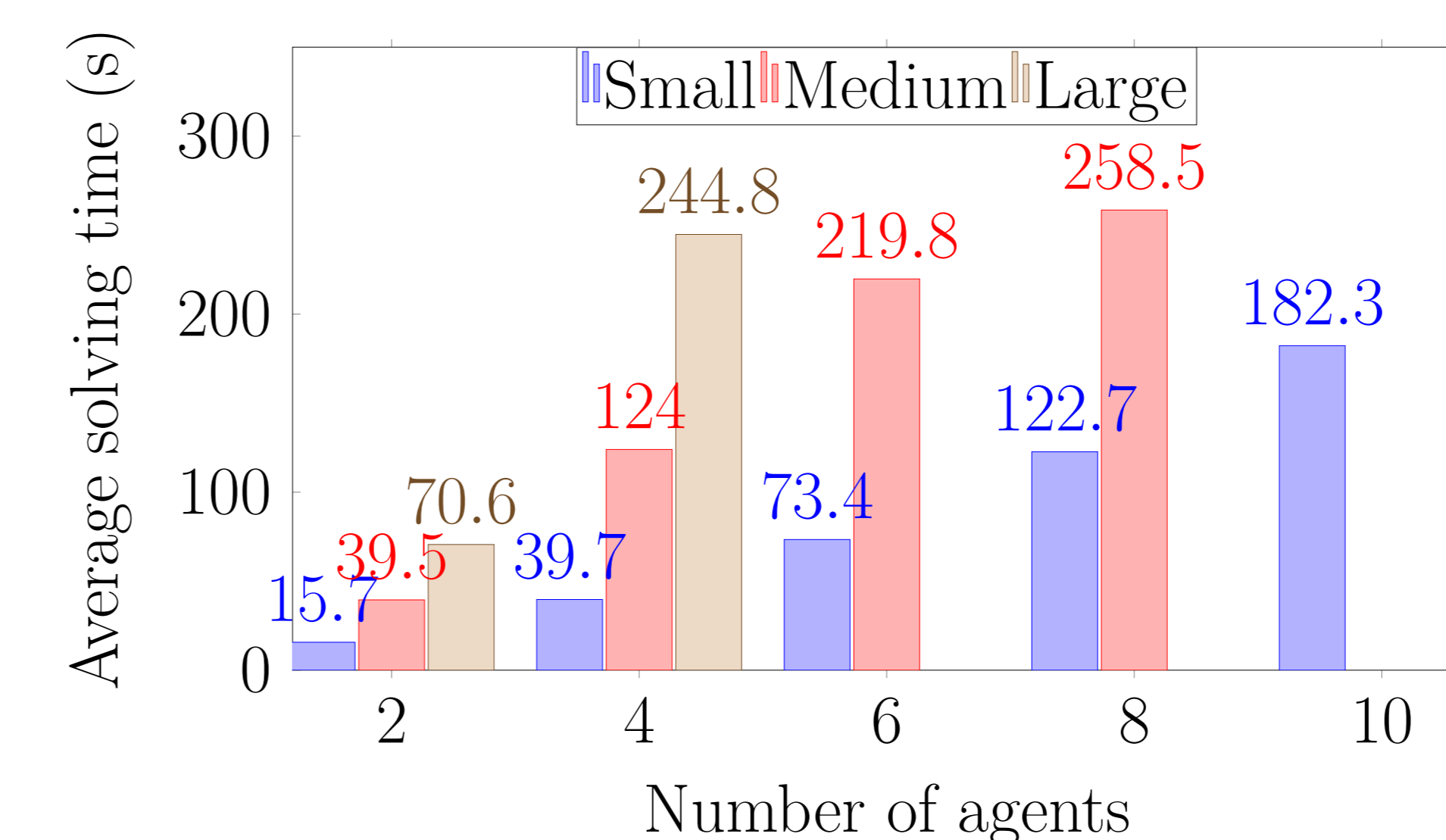


Figure: Average solving times for different combinations of maps and number of agents.

Conclusions

- Approach to Collective Adaptation Systems resilient to changes.
- Adaptation issues solved within an ensemble.
- Solve issues collectively with concurrent planning.

References

- [1] M. Fox and D. Long. PDDL2.1: an extension to PDDL for expressing temporal planning domains. *J. Artif. Intell. Res. (JAIR)*, 20:61–124, 2003.
- [2] J. Rintanen. Complexity of Concurrent Temporal Planning. In *ICAPS'07*, pages 280–287, 2007.
- [3] M. Haklay and P. Weber. OpenStreetMap: User-Generated Street Maps. *IEEE Pervasive Computing*, 7(4):12–18, 2008.
- [4] S. Jiménez, A. Jonsson, and H. Palacios. Temporal Planning With Required Concurrency Using Classical Planning. In *ICAPS'15*, pages 129–137, 2015.

Acknowledgments

This work has been partially funded by the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502).

Contact Information

- Software: <https://github.com/aig-upf/smart-carpooling-demo>
- Email: daniel.furelos@upf.edu

