PlanCurves: A Visualization Interface for Multi-Agent Temporal Plans

Pierre Le Bras[1], **Yaniel Carreno**[2], **Ronald P. A. Petrick**[1,2], **Mike J. Chantler**[1,2]
[1]Department of Computer Science, Heriot-Watt University, Edinburgh, UK. [2]Edinburgh Centre for Robotics, UK.
{p.le_bras, yc66, r.petrick, m.j.chantler}@hw.ac.uk

Abstract

Assessing the automatically-generated plans for operations and interactions between multiple agents over long periods of time can represent a challenging task for end-users. In many operational contexts, this verification and risk analysis step is crucial for safety and accountability. In this demonstration, we present PlanCurves, a visualization interface designed to not only show the planned actions of each agent, but also highlight their relative similarities across the plan. We show how this system is both planner and domain agnostic, and how it can adapt to user needs.

Introduction

Planning and scheduling systems aim at generating reliable and efficient plans. However, there are still situations where a human supervisor needs to evaluate plans and approve their implementation, for example: ensuring that the planned movements of mobile agents are accounted for by an operator. In this demonstration, we introduce PlanCurves to enhance the communication of plans to end-users by means of visualizations. In particular, we address the issue of multi-agent temporal plans, where a user would need to understand the planned operations of multiple agents across time.

While there have been plan visualizations systems implemented prior to this work, they often aim at representing domains for editing or explanation purposes. Our work focuses primarily on the representation of plan output. We combine a classical and familiar approach, Gantt charts, which align a set of planned activities on a time axis, with the less common use of *timelines*. Timelines have been used to show the evolution of variables specific to certain planners and domains. Here, we provide a novel implementation of timelines: using Time Curves (Bach et al. 2015), we allow the user to visualize the relative similarities between multiple agents' tracks, effectively creating an overview of the agents' interactions throughout the entire plan.

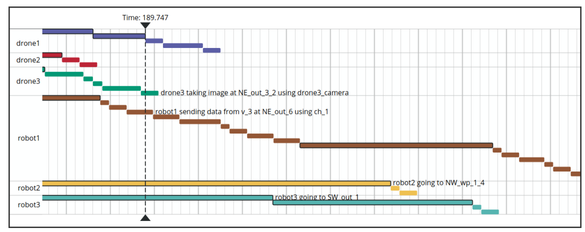


Figure 1: The Activity Chart depicts the sequence and timing of activities grouped by agent.

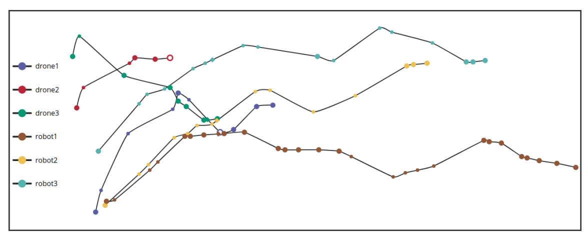


Figure 2: Time Curves depict the timeline (using similarity distortion) of multiple agents across the entire plan.

Visualizing Multi-Agent Temporal Plans

PlanCurves implements two main visualizations to display plans to users. Firstly, the Activity Chart (Figure 1) relies on the familiar principles of Gantt charts to display activities as horizontal bars, scaling their position and width to a time axis in order to convey two kinds of information: the sequence in which activities take place, and their timings. We also group the activities by actor, as it gives a better perspective of the overall agents' role and mission in the plan to the user, in multi-agent contexts in particular.

Secondly, we implement Time Curves, originally described by Bach et al. (2015) to represent similarity data of one entity across time. Here we apply this method to represent the actors' timelines simultaneously (Figure 2). The Time Curves present the timelines using a sequence of connected dots showing the actors' states throughout the plan. These timelines are however *distorted* to communicate the relative similarities between states. Using this type of visu-

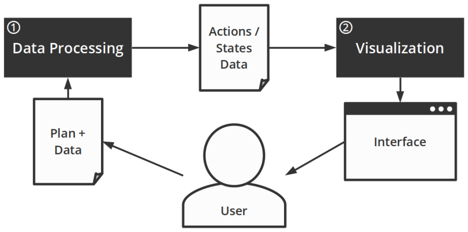


Figure 3: Flowchart of PlanCurves.

alization shows, in one display, the tasks of and *interactions* between actors. These interactions are based on the similarity measure used, which is a function of time and other features determined by the user, such as physical position in the environment or resource used.

The visualizations offer interactions within and between themselves. Inputs were also added to allow the user to modify the weights of features on Time Curves, offering other perspectives and supporting a deeper exploration of features.

System

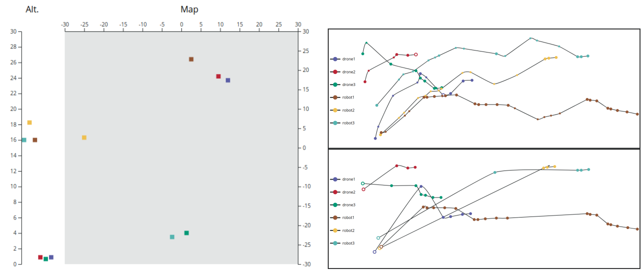
We have designed PlanCurves as an online application for practitioners to use in conjunction with any temporal planner and adapted to any domain specification that includes actors. The application only requires the user to load their plan along with data processing rules. The plan is processed to generate action and state data for the visualization, which are then printed on the interface (Figure 3).

The action data is directly generated from the plan file. The plan is first parsed to produce a list of actions separated in tokens to then interpret it. Two elements are mandatory for the interpretation: a) the first token is the action name, and b) the action has an actor token. The interpretation can be made automatically, the system, however, expects rules for each action type to be provided in order to fit the user's domain. We use this data to render the Activity Chart.

State data is then produced: from each action, the system infers a resulting state. While default inference rules are implemented, the user is able to customize these too. In particular, the user can specify the list of features for each state. Roadmap data may be provided, where point names are mapped to coordinates. These features will then be used to construct the state similarity matrix, used to produced the Time Curves by means of dimensionality reduction (Multi-Dimensional Scaling - MDS) (Brandes and Pich 2007). PlanCurves supports both numerical and categorical features, using the general similarity coefficient introduced by Gower (1971).

Additional Features

While we aimed at making PlanCurves domain agnostic, our initial use case emphasized the inspection of planned movements. As such our system includes functionalities to support environment analysis. First, we have implemented a Scene Map visualisation to simulate the planned position of robots throughout the plan (Figure 4a). Second, we have



(a) The Scene Map simulation. (b) Time Curves with Path Planning (top), and without (bottom).

Figure 4: Environment analysis features of PlanCurves

added an optional path planner based on A* search with a Euclidean heuristic (Hart, Nilsson, and Raphael 1968). With a list of waypoints in the roadmap data, the system estimates the route taken by an agent during *movement* actions labelled by the user (Figure 4b). This functionality complements the space representation that often limits planners.

Conclusion

The system described above constitutes the core of PlanCurves: an online, planner and domain agnostic, visualization system for practitioners and end-users to inspect and assess multi-agent temporal plans. It focuses on two main plan representations: an Activity Chart showing the listing and timing of proposed actions, and Time Curves presenting an overview of the agents' timelines distorted by their relative similarities to see their interactions.

In this demonstration we will present the PlanCurves system and detail the use and relevance of these visualizations to represent multi-agent temporal plans. We will also showcase its accessibility, ease of use, and adaptability.

Acknowledgements

This work was funded and supported by the ORCA Hub, under EPSRC grant EP/R026173/1.

References

Bach, B.; Shi, C.; Heulot, N.; Madhyastha, T.; Grabowski, T.; and Dragicevic, P. 2015. Time curves: Folding time to visualize patterns of temporal evolution in data. *IEEE transactions on visualization and computer graphics* 22(1):559–568.

Brandes, U., and Pich, C. 2007. Eigensolver methods for progressive multidimensional scaling of large data. In Kaufmann, M., and Wagner, D., eds., *Graph Drawing*, volume 4372 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. 42–53.

Gower, J. C. 1971. A general coefficient of similarity and some of its properties. *Biometrics* 857–871.

Hart, P. E.; Nilsson, N. J.; and Raphael, B. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE transactions on Systems Science and Cybernetics* 4(2):100–107.