CAST Instantiation of the NiFTi UGV Cognitive Architecture for planning, dialogue, functional mapping and high level control

Mario Gianni

December 17, 2013

1. We have designed a model-based control of the main components and processes of the NIFTi UGV robot [7]. The model is specified in the Temporal Flexible Situation Calculus (TFSC) which is a framework to suitable represent temporally-flexible behaviours in the Situation Calculus [5, 6]. This hybrid framework combines temporal constraints reasoning and reasoning about actions.

2. We have integrated the model-based control with incoming perceptual information from vision, SLAM, topological map segmentation and dialogue.

3. We have deployed an hybrid CAST subarchitecture which embeds the ECLIPSE Prolog implementation of the model-based control as well as a set of ROS nodes which are responsible of the communication with the ROS layers of the architecture of the NIFTi UGV robot. The CAST subarchitecture for the planning allows the human operator to switch between several operational modalities lying between autonomous and teleoperated modes during the execution of a task.

4. The planning subarchitecture provides communication interfaces with the human in both the directions: from the human to the robot and vice versa. The research related to representing tasks and responsibilities and authorizations in a shared display has been contributed by TNO as described in the work [4]. The research related to the situated dialogue processing and its integration into task-driven collaborative context as well as how the communication has been grounded in the human-robot collaboration has been contributed by DFKI as described in the work [8].

The model-based control of NIFTi UGV robot is composed of a TFSC model of the controllable activities and a planning engine. The TFSC model explicitly represents the main components and processes of the NIFTi UGV robot, the cause-effect relationships as well as the temporal constraints among the processes. These processes are represented in the TFSC through fluents and

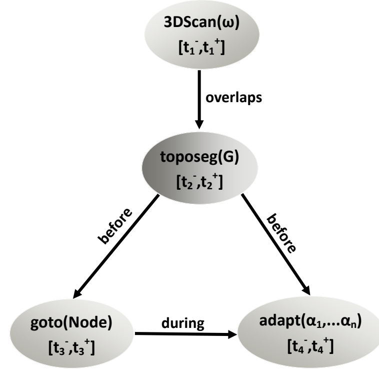

Figure 1: The network states that in order to reach the destination node $Node$ of the graph G, the system has to scan the environment, rotating the 3D laser with the angular velocity ω, segment the free space in order to build the graph G. During the execution of the navigation goal a suitable set of parameters α (e.g. flipper angles, velocity, acceleration) have to be available for the reconfiguration of the pose of the robot

instantaneous starting and ending actions which are defined in terms of preconditions and effects. Hard time constraints among the processes are managed by the TFSC model using Allen-like temporal relations [1].

The interactions between the set of the processes of the NIFTi UGV robot can be defined by a Temporal Constraint Network (TCN) [3], which represents the temporal relations among a set of parametric processes. Figure 1 shows a Temporal Constraint Network which defines a set of the temporal constraints on a subset of activities of the NIFTi UGV robot.

Temporal Flexible Situation Calculus intermediates between Situation Calculus formulae and Temporal Constraint Networks. Each temporal constraint between processes can be defined by a $\mathcal{L}_{TFSC}$ formula mentioning sets of timelines. The consistency of the temporal constraint network allows us to represent the set of temporal constraints on a subset of processes of our system with a consistent formula of the language $\mathcal{L}_{TFSC}$. The time variables mentioned in the formulae are assigned according to a specified set of behaviours, in so being very well suited for representing the flexible behaviours of the NIFTi UGV robot.

The planning engine is composed of two main logical modules: the plan generator and the execution monitoring. The plan generator relies on a library of Prolog scripts designating the set of tasks which the mobile robot can perform, according to the specified processes, their temporal constraints and preconditions. The tasks are based on the perceived information of the environment which is represented in the knowledge of the planning engine in the form of a graph augmented with information about the type and position of detected

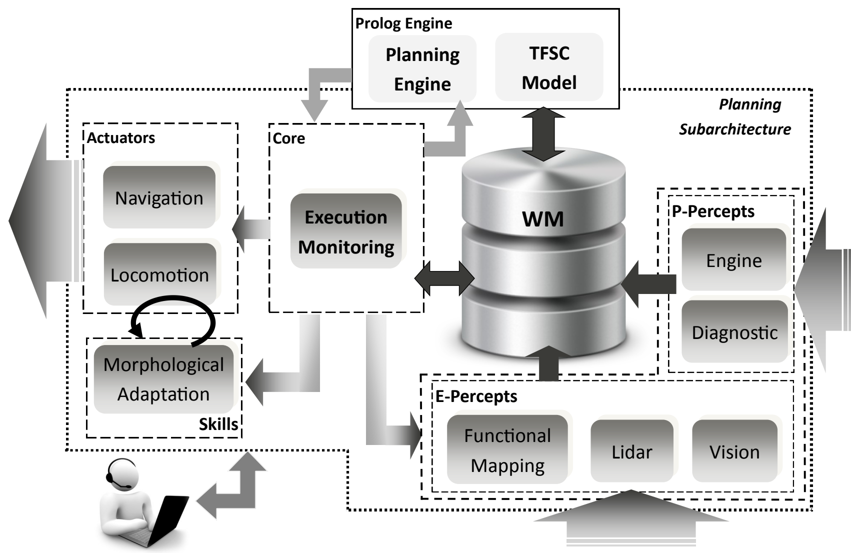

Figure 2: The planning subarchitecture

objects, within the map, such as cars or containers. The nodes of the graph correspond to topologically segmented regions or approachable areas around a detected object and edges determine the traversability between the regions.

The execution-monitoring ensures that the set of action sequences, generated by the plan generator, according to the TFSC model and the current state of the domain knowledge, are consistently executed. Furthermore the execution-monitoring manages the interventions of the human operator during the interaction with the control system.

Both the TFSC model and the planning engine are implemented in ECLIPSE Prolog [2] which optimally combines the power of a constraint solver with inference in order to generate the set of action sequences, and also enable the continuous update due to incoming new knowledge.

An hybrid CAST subarchitecture has been deployed in our mobile system in order to fully embed the TFSC model and the planning engine as well as a set of ROS nodes which are responsible of the main communication tasks with the ROS layers. Figure 2 illustrates the role of each component of the planning subarchitecture which we have designed for the robot.

The core of the planning subarchitecture is implemented by the *Execution-Monitoring* component which plays a crucial role in orchestrating the other components. This component manages the communication with the human interfaces and embeds into the subarchiteture the logical part of the control system. It enables the human operator to interact with the control system during the computational cycle. Further, this component allows the human operator to modify the control sequence produced by the planner by skipping some activities, adding new actions. It also allows the operator to take the control of some functional activities while the rescue robot is executing a task.

On the other hand the Execution-Monitoring component manages the external and internal *perceptual* components of the planning subarchitecture which communicate with the CAST subarchitectures/ROS modules implementing the perception capabilities of the NIFTi UGV robot. The information acquired from the *perceptual* components is written in the working memory *WM* and compiled in order to build the domain knowledge of the planning engine. The design of messages and data exchange coming in and out from the visual detector and associated with auxiliary ROS nodes has also involved CTU.

The component is also responsible for sending task activation signals to the *actuator* components of the subarchitecture in order to perform the sequences of actions generated by the planning engine.

The actuator components rely on the *skill* components to select the suitable set of parameters of the actions according to the internal state of the system and the features extracted from the 3D point cloud of the environment.

In other words the *morphological adaptation* generates sequences of parameters to configure the pose of the robot in order to reduce instabilities that could cause the tip-over. These sequences are for example flipper angles, velocity and acceleration. In order to generate the above defined parameters, the morphological adaptation component takes as inputs the 3D point cloud registered by the *lidar* component and the route computed by the *navigation* component to reach the destination point in the map.

References

[1] James F. Allen. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26(11):832–843, 1983.

[2] Krzysztof R. Apt and Mark Wallace. *Constraint Logic Programming using Eclipse*. Cambridge University Press, 2006.

[3] Rina Dechter, Itay Meiri, and Judea Pearl. Temporal constraint networks. *Artif. Intell.*, 49(1-3):61–95, 1991.

[4] Van Diggelen et al. Mutual empowerment in mobile soldier support. *Journal of Battle Technology (accepted)*, 2012.

[5] Alberto Finzi and Fiora Pirri. Representing flexible temporal behaviours in the situation calculus. In *In Proceedings of IJCAI*, 2005.

[6] Alberto Finzi and Fiora Pirri. Switching tasks and flexible reasoning in the situation calculus. Technical Report 7, Dipartimento di informatica e Sistemistica Sapienza Università di Roma, 2010.

[7] M. Gianni, P. Papadakis, F. Pirri, M. Liu, F. Pomerleau, F. Colas, K. Zimmerman, T. Svoboda, T. Petricek, G. J. M. Kruijff, H. Zender, and H. Khambhaita. A unified framework for planning and execution-monitoring of mobile robots. In *Proceedings of the AAAI-11 Workshop on Automated Action Planning for Autonomous mobile Robots*, 2011.

[8] G.J.M. Kruijff, M. Janíček, and H. Zender. Situated communication for joint activity in human-robot teams. *IEEE Intelligent Systems*, 2012. forthcoming.