28

Programming Bounded Rationality

Hans-Dieter Burkhard

Institute of Informatics
Humboldt University, D-10099 Berlin, Germany
hdb@informatik.hu-berlin.de

Summary. Research on Artificial Intelligence and Robotics helps to understand problems of rationality. Autonomous robots have to act and react in complex environments under constraints of their bounded resources. Simple daily tasks are much more difficult to implement than playing chess. Soccer playing robots are considered as test field for rational behavior. Our implementations are inspired by the belief-desire-intention model.

28.1 Introduction

Control of autonomous robots and agents in dynamical environments is interesting from a cognitive point of view as well as under application view points. Different architectures have been inspired by cognitive issues as well as technical requirements. It is commonly accepted that simple stimulus response behavior as well as deliberative decisions are both useful according to different requirements. Hybrid architectures are built to combine both.

The underlying conflict between complex computations for efficient behavior and bounded resources is known as bounded rationality. In his book [2], Bratman has investigated a model based on the concepts of belief (what the agent supposes the world to be), desires (what the agent desires, but not necessarily tries to achieve), and intentions (what the agent intends to achieve and the actions he wants to perform for that achievements). There is a sophisticated process of refinements of intentions and plans which finally lead to actions.

Implementations of this model are known as BDI architectures (BDI = belief, desire, intention, [7], [11]). Actually, the usage of the notions in the implementations is not unique. There were successful models in an multi-modal logical framework. BDI approaches are often identified with these models and their applications. In contrast, our approach uses the BDI concepts as bases for data structures in object oriented implementations.

The aim is to implement and maintain a data structure which corresponds to the idea of intentions as hierarchical partial plans which are completed at that time when it is necessary. We especially address problems known as upwards and recognizant failures: In hybrid architectures certain failures are recognized at the lower levels, but need handling on higher levels. As an example we use the soccer domain from the RoboCup initiative.

The author likes to thank the previous and recent members of the RoboCup teams "AT Humboldt" (Simulation League) and "German Team" (Sony Four Legged Robotic League) for a lot of fruitful discussions. The paper could not have been written without their theoretical and practical work. The work is granted by the German Research Association (DFG) in the main research program 1125 "Cooperating teams of mobile robots in dynamic and competitive environments".

28.2 Robot Control in Dynamic Environments

Control of autonomous robots in dynamic environments is often considered under the view point of moving vehicles with the need for fast short term reactions (e.g. obstacle avoidance) but sufficient time for long term planning. Layered architectures with simple "reactive" low level behavior and complex high level behavior are state of the art (cf. [1],[6]). Only the low level reactions are considered as time critical, while re-planning after the occurrence of unexpected events might take longer time. The vehicle must perform an immediate stop if the road is suddenly closed, but then it might wait until a new path is planned.

This need not to be true in other scenarios. Soccer can serve as an example for explanation. During an offensive, the players of team A are oriented forward. Suddenly, when the opponents get the control over the ball (e.g. after missing a pass by team A), then team A should immediately switch to defensive. What does it mean? Firstly, players should *immediately* stop running forward, they need not to run free anymore. Up to this point, it is comparable to the situation of a vehicle which stops moving before an obstacle. Secondly, it means *immediately* to adapt defensive play: Instead of running free (trying to keep distance to opponents) they have to mark (try to come closer to opponents and to attack them). There is no time for making new plans, the situation becomes different from the case of vehicles.

28.2.1 Soccer Robots: The RoboCup Initiative

The soccer scenario has been proven to be very useful for the discussion of problems and for the evaluation of proposed solutions for autonomous mobile robots in dynamic environments. Such environments are characterized by fast changes, plans may become invalid by unpredictable events. It is not possible to predict future events in general. Moreover, sensory information is incomplete and unreliable. Robots have to be aware, that their skills may not be successful and that their plans may fail. One may theoretically think about a plan to play the ball via several players from the goal-kick to the opponents goal, but nobody would expect that plan to work. Note that there is a great difference to a chess program: It is easy to write a program for finding the ultimate best moves, it is "only" a question of complexity to run this program. But nobody is able to write a similar program for soccer playing robots.