Integration of the ROS Framework in Soccer Robotics: the NAO Case

Leonardo Leottau Forero, José Miguel Yáñez, and Javier Ruiz-del-Solar

Advanced Mining Technology Center, Department of Electrical Engineering,
Universidad de Chile,
Av. Tupper 2007, Santiago, Chile
{dleottau,jruizd}@ing.uchile.cl, jm.yanez.arancibia@gmail.com
http://uchilert.amtc.cl

Abstract. The SPL robot soccer league focuses its efforts on the development of robot control software for standard humanoid robots. Nevertheless, few interchange of software modules are observed in the league, being the B-Human effort an exception. In addition, a large difference in performance is observed between experienced teams and new teams. This situation makes difficult the incorporation of new teams in the league. Therefore, it seems attractive to explore the use of ROS within the SPL soccer robotics community in order to revert the described situation. As a first step, this paper presents some work in this direction, such as the installation of ROS in the new NAO V4 robots, the integration of the B-Human motion engine as a ROS node, and the communication of two robots running a ROS-based control software.

Keywords: Humanoid robots, NAO robot, Robot Operating System (ROS), shared code.

1 Introduction

The SPL (Standard Platform League) is a RoboCup robot soccer league in which all teams compete with identical robots; since 2008 the robots being used are the Aldebaran NAO robots [4]. The research efforts focus on the development of sophisticated control software for fully autonomous operation of a team of robots. With teams participating in the league for 5-10 years, or ever more, some control software solutions have reached a very high level of effectiveness and functionality, which new teams cannot reach easily. In addition, even tough being a league where most of the work is on the development of software, there are very few cases of code sharing. One of these cases is the one of the B-Human team [1], which makes every year its software open to the community. However, even in this case it is not easy to use just some specific parts of the code. Normally teams use the whole code and modify the high-level behaviors. In addition, it is not easy for other teams to develop new components that can be integrated in the B-Human software. Basically, the B-Human software is being developed and extended just by the B-Human team.