Supporting Training on a Robotic Simulator using a Flexible Path Planner

Roger Nkambou[1] , Khaled Belghith[2], Froduald Kabanza[2], Mahie Khan[1]

[1] *Université du Québec à Montréal, Montréal (Québec) H3C 3P8, Canada*
nkambou.roger@uqam.ca ; khan.mahie@uqam.ca

[2] *Université de Sherbrooke, Sherbrooke (Québec) J1K 2R1, Canada*
khaled.belghith@USherbrooke.ca ; kabanza@USherbrooke.ca

Abstract. Manipulating the Space Station Remote Manipulator (SSRMS) on the International Space Station (ISS) is a very challenging task. The operator does not have a direct view of the scene of operation and must rely on cameras mounted on the manipulator and at strategic places of the environment where it operates. In this paper, we describe how a new approach for robot path planning called FADPRM can be used to support the training of astronauts on such a manipulator and under this big constraint of restricted sight.

Introduction

We are engaged in a research project for the development of an intelligent tutoring system called *Roman Tutor* to support astronauts in learning how to operate the SSRMS, an articulated robot arm mounted on the international space station (ISS). Astronauts operate the SSRMS through a robotic workstation located inside one of the ISS compartments. This workstation is equipped with an interface containing three monitors, each connected to one of the fourteen cameras placed at a strategic location on the ISS.

The SSRMS can be involved in various tasks that must be carried out very carefully on the ISS, ranging from moving a load to inspecting the ISS structure and making repairs. At different phases of a given manipulation, the astronaut must choose a setting of cameras that provides him with the best visibility while keeping a good appreciation of his evolution in the task. As most complex tasks deal in one way or another with moving the SSRMS, for the simulator to be able to understand students' operations in order to provide feedback, it must itself be aware of the space constraints and be able to move the arm by itself. A path-planner that calculates arm's moves without collision and consistent with best available cameras view is the key training resource on which other resources and abstract tutoring processes hinge. In this paper we describe how the FADPRM path-planner can be useful in providing amazingly tutoring feedback to a student (astronaut) during telemanipulation activities.

1. The FADPRM Path-Planner

In the literature, several approaches dealing with the path-planning problem for robots in constrained environments were found. Several implementations were carried out on the basis of these various approaches and much of them are relatively effective and precise. The fact is that none of these techniques deals with the problem of restricted sight we are faced with in our case. That's why we designed and implemented a new flexible and effective approach for robot path planning we call FADPRM [1].

FADPRM is a combination of the traditional PRM approach and AD*. It is flexible in that it takes into account zones with different degrees of desirability.

More specifically, FADPRM allows putting in the environment different zones with arbitrary geometrical forms. A degree of desirability *dd*, a real in [0 1], is assigned to each zone. On the ISS, the number, the form, the *dd* and the placement of zones reflect the disposition of the cameras on the station. The *dd* of a desired zone, such as a zone covering the field of vision of a camera, is then positive and the more it approaches 1, the more the zone is desired; the same for a nondesired zone where the *dd* is near 0. Thus, the FADPRM path-planner will try to bring the robot in zones offering the best possible visibility of the progression while trying to avoid zones with reduced visibility.

2. The Roman Tutor

The *Roman Tutor* user interface (Fig. 1) simulates that of the real workstation on which the astronauts operates the SSRMS. It contains three monitors with some buttons and functionalities to move the corresponding cameras: Tilt, Pan and Zoom.

The *Trace window* at the bottom of the *Roman Tutor* keeps a continuous track of all the operations done so far by the learner (the selection of a new camera in a monitor, the move of a camera and of the robot) and contains all information about the current state of the robot (if there is a collision or not, End-Effector's coordinates, cameras' positions etc.).

While the learner is carrying on some tasks on the simulator, traces of his progression are stored in files. The *Tutor* then parses these files in order to analyze the knowledge acquired by the learner, and to diagnose exactly where the gaps are. The *Tutor* diagnosis is done based on the plans generated by the FADPRM Planner and also on domain knowledge related to these plans. ISS and SSRMS domain knowledge is modeled as a Bayesian network which shows causal relationships between knowledge. *The Tutor* relies on this structure to diagnose students' errors in terms of lack of knowledge, misconception, etc.

The *Tutor* can also choose to provide the learner with a partial or total illustration of the task he's working on by calling the *Animation Generator*. The *Task Generator* allows the domain expert to design new tasks on which we might want to train astronauts.

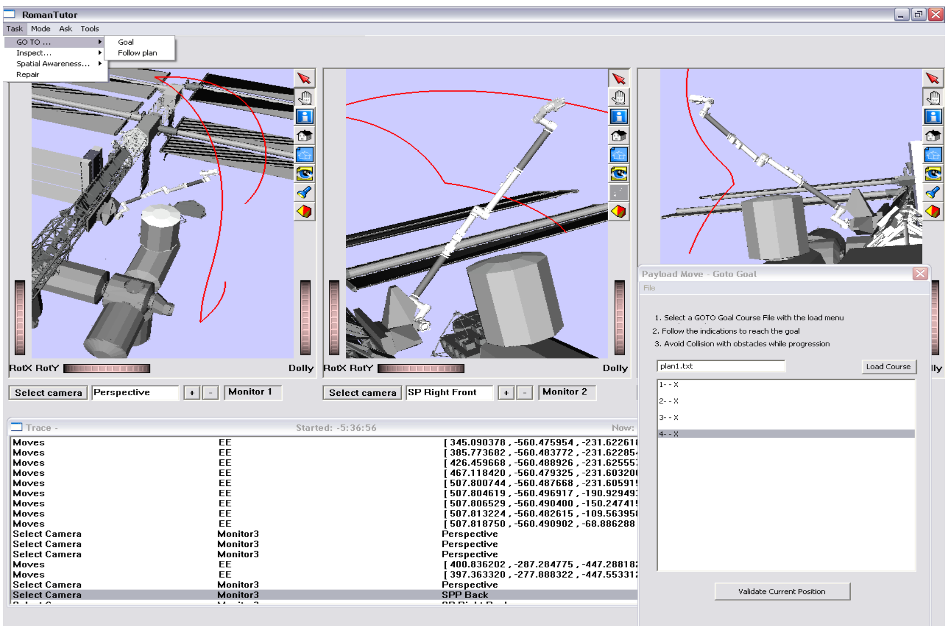

Figure 1. Roman Tutor User Interface (GoTo Task)

3. Using FADPRM Path-Planner for the Tutoring Assistance

One of the main goals of an intelligent tutoring system is to actively provide relevant feedback to the student in problem solving situations [2]. This kind of support becomes very difficult when an explicit representation of the training task is not available. This is the case in the SSRMS environment where the problem space associated with a given task consists of an infinite number of paths. *Roman Tutor* overcomes this problem by using FADPRM as principal resource for the tutoring feedback. *Roman Tutor* includes four different types of tasks: Spatial Awareness, GoTo, Inspect and Repair. The 'Spatial Awareness' task, improves the learner's knowledge about the space station's environment by providing him with some exercises such as naming and locating ISS elements, zones and cameras. In the 'GoTo' task (Fig. 1), the learner uses the SSRMS to move a load from one position to another different. The *Tutor* then executes the FADPRM planner, which generates a plan (a sequence of points) that joins the two positions. Based on this plan, the *Tutor* can: validate student action or sequence of actions, give information about the next relevant action or sequence, and generate relevant task demonstration resources.

During the evolution of the astronaut in a task, *Roman Tutor*, by calling the *Movie Generator* component, might choose to provide him with an animation illustrating entirely or partially the task he has to do. The movie generated after a call to the FADPRM path-planner, takes into account the disposition of the cameras on the station. In fact, it is constituted of a series of sequences taken from different and appropriate cameras showing the displacement of the SSRMS. For each sequence in the plan, the camera that gives the better sight of the displacement of SSRMS is chosen.

The learner is also provided with the 'Ask' menu, which allows him to ask different types of questions while executing a task. These questions may be of three different forms: How-To, What-If and Why-Not. *Roman Tutor* answers How-To questions by generating a path using FADPRM and by building an interactive animation that follow that path (as explained above). The incremental planning capability of FADPRM is used by *Roman Tutor* to bring answers to the What-If and Why-Not questions. In both cases, *Roman Tutor* provides the learner with relevant explanations given that his action or sequence of actions is out of scope of the generated plan or may bring him in a dead end.

4. Conclusion

In this paper, we described how a new approach for robot path planning called FADPRM could play an important role in providing tutoring feedbacks to the learner during training on a robot simulator. FADPRM is integrated into *Roman Tutor* allowing it to provide the learner with a continuous and relevant assistance during his progression in the task. Many tests and experiences have been carried out and we obtained very good and satisfactory results. In fact, we noticed that FADPRM enhances considerably the efficiency of the tutoring and improves the training quality. This constitutes a very important contribution in the field of intelligent tutoring systems. In fact, our results indicate that it is not necessary to explicitly create a complex problem space or task graph to support the tutoring process.

References

[1] F. Kabanza, R. Nkambou, K. Belghith, "*Path-Planning for Autonomous Training on Robot Manipulators in Space*". In Proc. of 19th International Joint Conference on Artificial Intelligence (IJCAI), 2005.
[2] K.VanLehn, "*The advantages of Explicity Representing Problem Spaces*". User Modeling 2003, Springer Verlag LNAI 2702:3.