EE4306: DISTRIBUTED AUTONOMOUS ROBOTIC SYSTEMS
2014/2015 SEMESTER 2
V-REP AND EXTERNAL CONTROLLER: LAB 2

OBJECTIVES

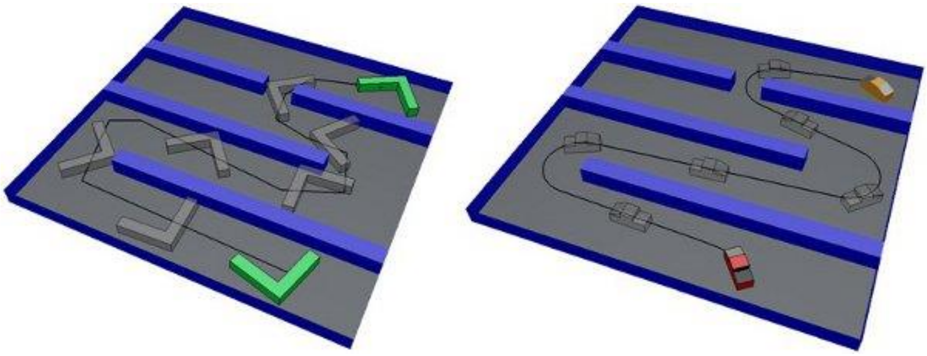
This lab familiarises you the use of Virtual Robot Experiment Platform (V-REP) with embedded script or external controller. The objectives of the lab are

1. Implement a pathfinding algortihm.
2. Develop a strategy for obstacles avoidance.
3. Develop locomotion for differential drive steering.

You have to answer question 1 (***Q.1***) in this manual by developing pathfinding robot. The grading will be based on the accuracy of your program and report.

INTRODUCTION

V-REP's path planning module allows handling path planning tasks in 3D-space, and in 2D-space for vehicles with non-holonomic motion constraints. The path planning module does not include motion planning for kinematic chains, which is handled by the motion planning module. Following figures illustrate some path planning examples in V-REP:



(1) holonomic path planning, (2) non-holonomic path planning

A path planning task usually takes several input values or parameters:

- **a start position (or start configuration):** this is the initial configuration of a device (e.g. robot).
- **a goal position (or goal configuration):** this is the desired configuration for the device or robot.
- **obstacles:** those are the objects that the device (or robot) shouldn't be colliding with, while following a path from the start to the goal configuration.

A path linking the start configuration to the goal configuration can be specified (or restricted to be) in a configuration space with a specific number of dimensions (e.g. the X, Y configuration space). Moreover additional constraints are usually needed that make the task more complicated (e.g. keeping a certain distance threshold to the obstacles, or moving only in one direction).

SECTION A: V-REP SIMULATION ENVIRONMENT

LEARNING OUTCOMES:

(i) Tracking environment condition.
(ii) Develop pathfinding algorithm

1. Download and open ***LAB2_Environment.ttt*** from IVLE workbin. In the simulated scene you will find model ***ePuck*** mobile robot.

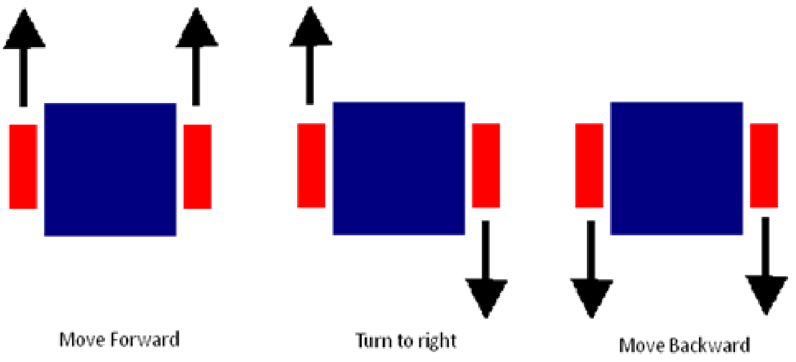
Actual ePuck mobile robot (Source: www.e-puck.org)

2. Run the simulation using button and observe the trajectory of the ***ePuck.***
3. Stop the simulation by clicking button.
4. The remoteAPI connection is initialised in ***Connection*** child script line 42. The socket communication port is set to ‘19999’.

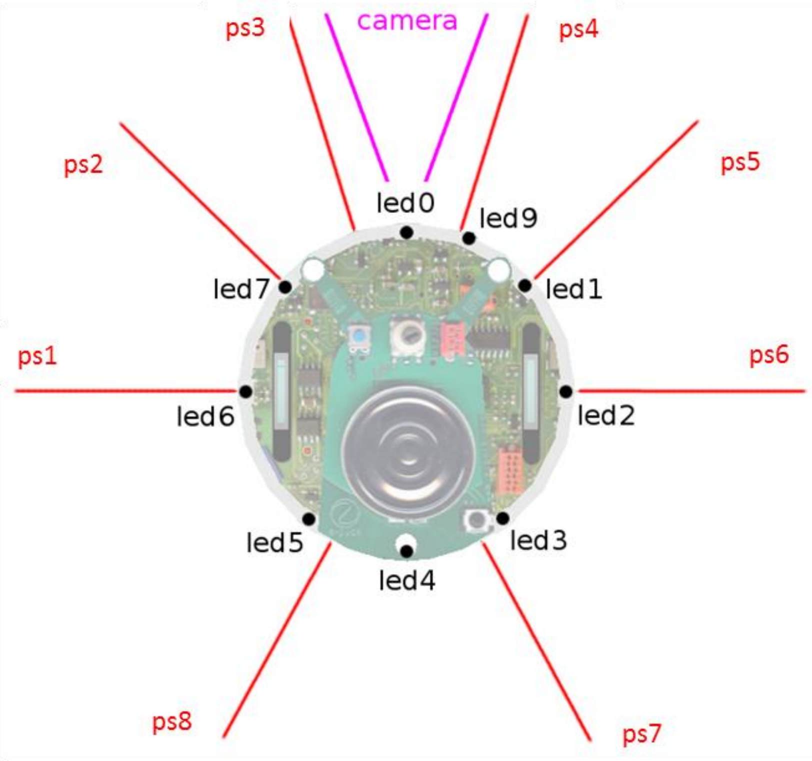
```
simExtRemoteApiStart(19999)
```

QUESTION 1 (10 marks)

The ***ePuck*** mobile robot has been developed as an open tool. The robot is integrated with many sensors such as proximity sensors and vision camera. The overall goal of this lab is to develop an algorithm to enable the ePuck to navigate from Start to Goal position without coming in contact with obstacles within the playing field. ***Puck*** is a 2 wheeled differential drive robot with 8 infrared proximity sensors and camera to assist its navigation. The diagrams below show ***ePuck*** a differential drive and its sensor locations.

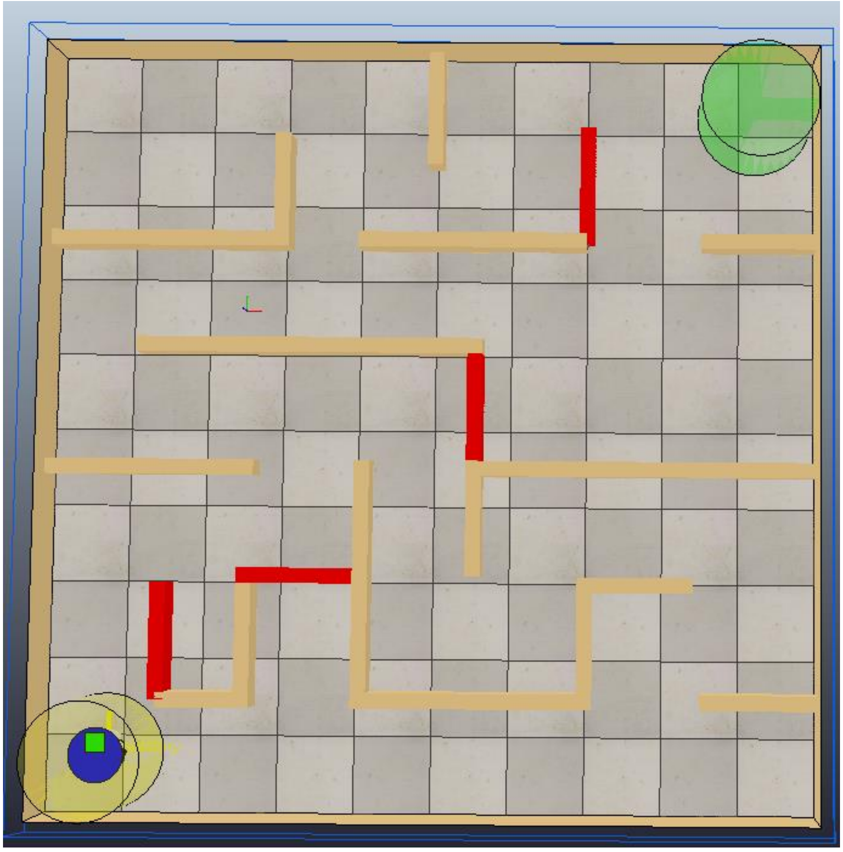


Differential drive



ePuck sensors diagram

In additional to the on board sensors, the play field is also equipped with a camera (***Global_Vision***) which give a plan view of the entire environment. The **YellowZone** (lower left) and **GreenZone** (lower left) shown in the diagram below are the start and goal location respectively. The **Red Cuboids** are door like obstacles that will open and close at different time in the simulation. The objective of the lab is as follows:



ePuck mobile robot in the simulation environment

Objectives:

(i) Develop a locomotion strategy for an ***ePuck*** differential drive. (2 marks)
   a. Solving wheel velocity with relation to heading of ***ePuck***

(ii) Develop an algorithm to navigate ***ePuck*** to **GreenZone** (goal location) (4 marks)
   a. To navigate to the goal location

(iii) Obstacle avoidance (3 marks)
   a. ***ePuck*** should not come in contact with any object in the environment

(iv) Record a video of the simulation and include the following (1 mark)
   a. ***ePuck*** begin in **YellowZone** and end in **GreenZone**
   b. ***ePuck*** navigates around the obstacles to the goal location
   c. Preferred format MP4/MPEG-4

You are to create your own code to achieve the above objectives, you may use and V-REP regular or remote APIs. Your marks will be awarded based on your approach and the capability of your program as well as its performance.

SUBMISSION

You are required to upload your submission to **IVLE,** should be submitted by 21st Feb 2015 (2359hrs). Report and codes will be combined and zipped, rename the zip file using your student number (i.e. A1234567Z_LAB1.zip) and upload the file to **IVLE–EE4306/ Workbin Project and Lab/Lab 1 Submission**. Video will be uploaded to another IVLE folder.

REPORT: The entire report should concise description of what you did, why, and what happened. The entire report should be no more than 2 pages (excluding appendices) and upload the in PDF format. It should contain the following sections:

1. Introduction: An overview of the main ideas used in your approach.
2. Algorithm and Strategy: Explain the techniques and path planning algorithm that you used. How are the ideas implemented. Where suitable, justify your decisions, e.g. why you used one method rather than another, what you tried that didn't work as expected, etc.
3. Results and Discussion: How well your algorithm performs, document the success and failure. Assess the success of your program with regard to the reported results, and explain any limitations, problems or improvements you would make.

CODE/PROGRAM: Save all codes and zip your project workspace folder. Remember to include a readme.txt file that describes how to run the program.

VIDEO: Rename the video file using your student number (i.e. A1234567Z_LAB2_Video.mp4) and upload the file to **IVLE–EE4306/ Workbin Project and Lab/Lab 2 Video**. (Note: File size may be large, you may want to upload the file within NUS network)

PLAGIARISM

This submission should be of your own work. Please read and understand the university policy on plagiarism: http://www.eng.nus.edu.sg/ugrad/SI_plagiarism.html

HELPFUL LINKS

You might wish to access the following links to gain some more understanding of the V-REP. Accessing them before this lab will make this lab very easy, and further accessing them after the lab will help you create better programs for your project:

V-REP User Manual:
http://www.coppeliarobotics.com/helpFiles/

Embedded Script:
http://www.coppeliarobotics.com/helpFiles/en/scripts.htm

Learning Lua Language by Derek Banas:
https://www.youtube.com/watch?v=iMacxZQMPXs

Learning Python Language by Derek Banas:
https://www.youtube.com/watch?v=N4mEzFDjqtA

Regular API function list:
http://www.coppeliarobotics.com/helpFiles/en/apiFunctionListAlphabetical.htm

Remote API function list:
http://www.coppeliarobotics.com/helpFiles/en/remoteApiFunctionListAlphabetical.htm

Youtube Tutorial, RemoteAPI Python by Nikolai K:
http://youtu.be/SQont-mTnfM

Evolutionary artificial potential fields and their application in real time robot path planning:
http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=870304&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D870304
or
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.125.7453&rep=rep1&type=pdf