Great. Very clear.

Robotics II
dVC Homework 1

Assigned: January ??, 2011
Due: January ??, 2011

I have a couple suggestions & found a couple typos.

Part 1: dVC Scene Basics

Deliverables:

1. Modified H1Part1A scene folder
2. New H1Part1D scene folder

Will they be using cygwin on windows machines? Does .net support this?

Introduction:

In dVC, a scene is a set of starting conditions and operating parameters for a simulation. For part 1 of the homework, we will be going over the basics of working with dVC and creating your own scene. On the course website, please download H1Part1A.zip. In this folder are the parts for a basic scene. Copy this folder into dvc-2D/examples. (I.E. dvc-2D/examples/H1Part1A)

dVC uses the CMake build system. In order for CMake to know what files and directories to use, and what to do with them, Files called CMakeLists.txt are used.

Sections:

A) Modify CMakeLists.txt so H1Part1A is available in dVC Qt.

To make sure our new scene is included in dVC, we need to modify the **CMakeLists.txt** file located in **dvc-2D/examples** to include our new directory. Open the file and locate the section that looks similar to this:

Listing 1: CMakeLists.txt

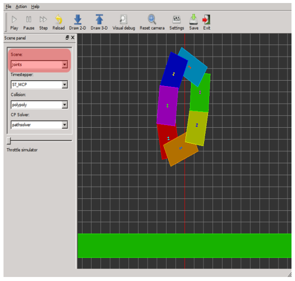
```
## Build the plugins examples.
if (BUILD_QT_APP)
  #add_subdirectory(plugins)
  add_subdirectory(benpqp_debugger)
  add_subdirectory(boxstack)
  add_subdirectory(circletest)
  ...
```

is this an executable?

Add the line `add_subdirectory(H1Part1A)` to this section. Once this change is made, rebuild dVC using make.

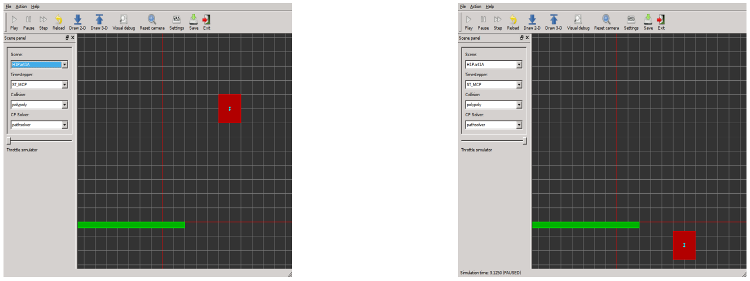
Open dvcQt. You should see something similar to Figure 1. In the Scene Selection drop-down menu, you should now have available **H1Part1A** as an option. Select this scene, you should then be presented with something that looks like Figure 2.

When you press the play button, you will notice the red block falls past the green platform down to infinity. (To make this easier to see, you can use the “Throttle simulator” slider.)



Scene: Joints
Timestepper: ST_MCP
Collision: polypoly
CP Solver: pathsolver

Figure 1: Opening dVCQt Scene, with Scene Selection drop-down highlighted



Scene: H1Part1A
Timestepper: ST_MCP
Collision: polypoly
CP Solver: pathsolver

Simulation time: 3.1250 (PAUSED)

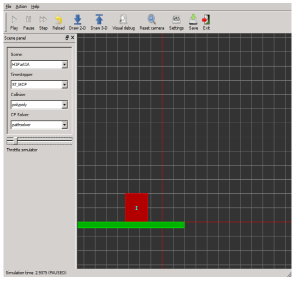
(a) Scene loaded

(b) Scene running

Figure 2: dvcQt Scene H1Part1A

B1) Modify H1Part1A's XML file to make the block fall onto the platform.

All the configuration parameters of the scene are located in an XML file. The XML file in the scene folder copied in section A is called `H1Part1A_prefs.xml.in`. Locate the geometry section (`<pref name='geometry'>`). In this section are all the properties of the objects in the scene. What you will want to modify for this step is the **Initial Position** of the **Box** so that it falls on the platform, similar to Figure 3



Scene: H1Part1A
Timestepper: ST_MCP
Collision: polypoly
CP Solver: pathsolver

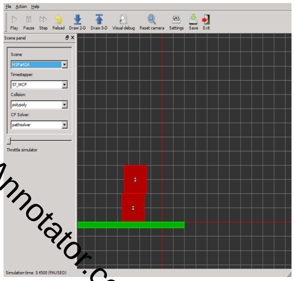
Simulation time: 2.9375 (PAUSED)

Figure 3: dVCQt Scene H1Part1A, Section B1, Box on platform

(Note:Modifying parameters like initial position, velocity, mass, etc. can be done quickly by clicking the "Settings" button in the toolbar. After making changes, press Apply, then Reload the scene)

B2) Modify H1Part1A's XML file to make a second block fall onto the first block on the platform.

In the same section of the XML file as section B1, you will now add a second block, which will fall on the first block, onto the platform. Most of the code can be copied and pasted, changing only the position and name (You can change the color to make things clearer, but it isn't necessary). It should look something like Figure 4.



Scene: H1Part1A
Timestepper: ST_MCP
Collision: polypoly
CP Solver: pathsolver
Simulation time: 3.4500 (PAUSED)

Figure 4: dVCQt Scene H1Part1A, Section B2, 2 Stacked Boxes

C1) Make a new geometry object in the shape of a pentagon.

vertices

Geometry objects for our scene are located in the H1Part1A/Geometry folder. Geometry files have a simple format. The first number in the top is the number of vertexes the object has. Next are the X and Y coordinates, specified clockwise. Looking at an existing geometry file should make it clear.

Create a new geometry file called `hw1_pentagon.pdat`, which has 5 sides (doesn't need to be a regular pentagon).

C2) Modify H1Part1A's CMakeLists.txt file to include your new geometry file.

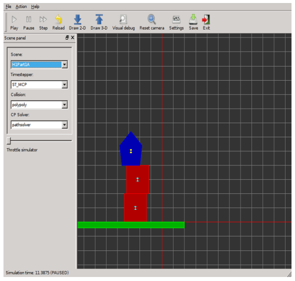
In the H1Part1A/Geometry folder is a CMakeLists.txt file that specifies which geometry files should be added to the dVC build. Add your new geometry file to the section `set(APP_GEOMETRY)` ?

C3) Add your new object to the scene so it falls on the platform on top of the two boxes.

?

Modify the XML file, creating a new object (Like in section B2), making sure to also modify `<pref name='vert_data_file'>` to use the new object you created. You should also adjust the `geomToBodyTransform` values, which specifies the location of the center of gravity (This only needs to be approximately correct). You should get something that looks like Figure 5

You can put in outside the body to make strange effects if you like.

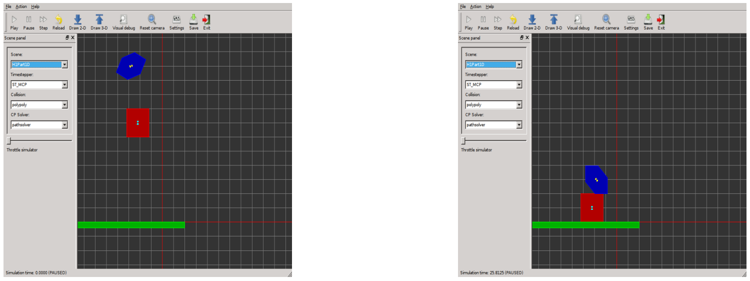


Scene: H1Part1A
Timestepper: ST_MCP
Collision: polypoly
CP Solver: pathsolver
Simulation time: 11.3875 (PAUSED)

Figure 5: dVCQt Scene H1Part1A, Section C3, Pentagon Stacked on boxes

D) Using what you learned above, create a new scene folder called H1Part1D. In this scene, have a square fall on the platform, and a hexagon fall on the square. The end result should look something like Figure 6.

Notes: – You can copy the H1Part1A folder, just make sure to rename the folder, XML, and Geometry. – The name of the scene appears 2 times in the XML file



Scene: H1Part1D
Timestepper: ST_MCP
Collision: polypoly
CP Solver: pathsolver
Simulation time: 0.0000 (PAUSED)

(a) Scene loaded

Scene: H1Part1D
Timestepper: ST_MCP
Collision: polypoly
CP Solver: pathsolver
Simulation time: 25.8125 (PAUSED)

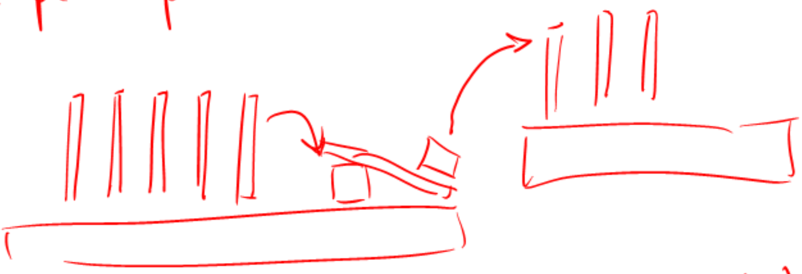
(b) Objects at rest

Figure 6: dVCQt Scene H1Part1D, Section D, A possible solution

****EXTRA CREDIT****

The most creative stacking of the box and hexagon will receive extra credit. You may use any capability of dVC to accomplish this. Changing the initial velocities, mass, and moment of inertia are good starting points. You may also add new objects to the scene, as long as there is a hexagon stacked on top of a box, resting on the platform.

They could perhaps build a dominos or Rube Goldberg machine.



Maybe give a couple of suggestions for how to get the extra credit.