Massachusetts Institute of Technology

16.410-13 Principles of Autonomy and Decision Making

Jump Starting lpsolve in Java

Introduction

This jumpstart shows you the basics of using a Linear Program solver, lpsolve, in Java. Please note that this jumpstart will give you only the most rudimentary skills in using lpsolve. Refer to the lpsolve API for more detail.

Download lpsolve

Download the following files from http://sourceforge.net/projects/lpsolve:

- lp_solve_5.5.0.13_dev.zip
- lp_solve_5.5.0.13_exe.zip.
- lp_solve_5.5.0.13_java.zip (JAVA wrapper)

The files with "dev" and "exe" include dynamic libraries and binaries of lpsolve, and the file with "java" includes a jar file and a dynamic library file that provide Java interface to lpsolve. The file with "java" also contains a doc folder in which the API can be found.

On mac with darwinports you can try "sudo port install lp_solve". Without darwinports, you should follow the generic linux instructions. Similarly, on a platform with aptitude (e.g. ubuntu) you can install with apt-get.

Install lpsolve

The following installation instruction has been extracted from the README.html in lp_solve_5.5.0.13_java.zip.

1. Copy the lp_solve dynamic libraries from the archives lp_solve_5.5_dev.(zip or tar.gz) and lp_solve_5.5_exe.(zip or tar.gz) to a standard library directory for your target platform.
   - On Windows, a typical place would be \WINDOWS or \WINDOWS\SYSTEM32.
   - On Linux, a typical place would be the directory /usr/local/lib.
   - If you installed with an automated tool (e.g. port or apt-get), this step should have been done for you.
2. Unzip the Java wrapper distribution file to new directory of your choice.
3. Copy the Java wrapper library into a standard library directory:
   - On Windows, copy the wrapper stub library lpsolve55j.dll to the directory that already contains lpsolve55.dll.

- On Linux, copy the wrapper stub library liblpsolve55j.so to the directory that already contains liblpsolve55.so. Run ldconfig to include the library in the shared library cache.

4. Copy the archive file `lpsolve55j.jar` from the Java wrapper distribution to a directory that is included in the CLASSPATH of your java program.
   - Alternatively, you can also select any directory of your choice (.e.g. C:\Program Files\lpsolve) and add that directory to the CLASSPATH.

Using lpsolve within Eclipse

If you wish to use lpsolve within eclipse environment, you must add the lpsolve library to your project. You can add lpsolve library to a project within the properties menu of the project. Under "Java Build Path" section, in "Libraries" tab, click on the "Add External JARs" button. Find the location of "lpsolve55j.jar" and click ok.

Simple Example

The following has been extracted from the README.html in lp_solve_5.5.0.13_java.zip.

$$\begin{aligned} &\text{Minimize} && z = 2x_1 + 3x_2 - 2x_3 + 3x_4 \\ &\text{Subject To} && 3x_1 + 2x_2 + 2x_3 + 1x_4 \le 4 \\ & && 4x_2 + 3x_3 + 1x_4 \ge 3 \end{aligned}$$

```
import lpsolve.*;
public class Demo {
  public static void main(String[] args) {
    try {
      // Create a problem with 4 variables and 0 constraints
      LpSolve solver = LpSolve.makeLp(0, 4);

      // add constraints
      solver.strAddConstraint("3 2 2 1", LpSolve.LE, 4);
      solver.strAddConstraint("0 4 3 1", LpSolve.GE, 3);

      // set objective function
      solver.strSetObjFn("2 3 -2 3");

      // solve the problem
      solver.solve();

      // print solution
      System.out.println("Value of objective function: " +
solver.getObjective());
      double[] var = solver.getPtrVariables();
      for (int i = 0; i < var.length; i++) {
        System.out.println("Value of var[" + i + "] = " + var[i]);
      }

      // delete the problem and free memory
      solver.deleteLp();
    }
    catch (LpSolveException e) {
       e.printStackTrace();
```

```
        }
    }
}
```

☺ Have Fun with lpsolve!!! ☺

16.410 / 16.413 Principles of Autonomy and Decision Making

Fall 2010