Using JAVA Applets

to

Teach Linear Programming

Robert J. Vanderbei

November 7, 2006

INFORMS’2006 — Pittsburgh

Work supported by ONR and NSF

http://www.princeton.edu/~rvdb
Linear Programming: Foundations and Extensions

- Order
- Table of Contents
- Lecture Notes - Undergraduate level (pdf format)
- Lecture Notes - Graduate level (pdf format)
- Online version of 2nd Edition (note that the hyperlinks in the text really work)
- Associated software (C/C++ source code)
- Additional Exercises
- Online Exercises
- Errata
- Simple pivot tool (a java applet to help solve exercises from Chapters 1 to 5)
- Advanced pivot tool (a java applet to help solve exercises from Chapters 5 to 7)
- Online pivot exam (a java applet to evaluate your knowledge of the simplex method)
- The Network-Simplex Pivot Tool
The Primal Simplex Method

maximize \( c^T x \)
subject to \( Ax \leq b \)
\( x \geq 0 \).

Rewrite with slack variables

maximize \( \xi = c^T x \)
subject to \( w = b - Ax \)
\( x, w \geq 0 \).

Solve with Pivot Tool

<table>
<thead>
<tr>
<th>obj = 0.0</th>
<th>+</th>
<th>-1.0</th>
<th>x1 +</th>
<th>3.0</th>
<th>x2 +</th>
<th>-3.0</th>
<th>x3</th>
</tr>
</thead>
<tbody>
<tr>
<td>w1 = 7.0</td>
<td>-</td>
<td>3.0</td>
<td>x1</td>
<td>-1.0</td>
<td>x2</td>
<td>-2.0</td>
<td>x3</td>
</tr>
<tr>
<td>w2 = 3.0</td>
<td>-</td>
<td>-2.0</td>
<td>x1</td>
<td>-4.0</td>
<td>x2</td>
<td>4.0</td>
<td>x3</td>
</tr>
<tr>
<td>w3 = 4.0</td>
<td>-</td>
<td>1.0</td>
<td>x1</td>
<td>0.0</td>
<td>x2</td>
<td>-2.0</td>
<td>x3</td>
</tr>
<tr>
<td>w4 = 8.0</td>
<td>-</td>
<td>-2.0</td>
<td>x1</td>
<td>2.0</td>
<td>x2</td>
<td>1.0</td>
<td>x3</td>
</tr>
<tr>
<td>w5 = 5.0</td>
<td>-</td>
<td>3.0</td>
<td>x1</td>
<td>0.0</td>
<td>x2</td>
<td>0.0</td>
<td>x3</td>
</tr>
</tbody>
</table>
Two-Phase Methods: Advanced Applet

An Example:

<table>
<thead>
<tr>
<th>obj</th>
<th>0.0</th>
<th>+</th>
<th>-4.0</th>
<th>x1</th>
<th>+</th>
<th>2.0</th>
<th>x2</th>
<th>+</th>
<th>3.0</th>
<th>x3</th>
</tr>
</thead>
<tbody>
<tr>
<td>w1</td>
<td>0.0</td>
<td>+</td>
<td>1.0</td>
<td>-</td>
<td>2.0</td>
<td>x1</td>
<td>-</td>
<td>-1.0</td>
<td>x2</td>
<td>-</td>
</tr>
<tr>
<td>w2</td>
<td>0.0</td>
<td>+</td>
<td>1.0</td>
<td>-</td>
<td>3.0</td>
<td>x1</td>
<td>-</td>
<td>-3.0</td>
<td>x2</td>
<td>-</td>
</tr>
<tr>
<td>w3</td>
<td>-3.0</td>
<td>+</td>
<td>1.0</td>
<td>-</td>
<td>-1.0</td>
<td>x1</td>
<td>-</td>
<td>-1.0</td>
<td>x2</td>
<td>-</td>
</tr>
<tr>
<td>w4</td>
<td>-1.0</td>
<td>+</td>
<td>1.0</td>
<td>-</td>
<td>-2.0</td>
<td>x1</td>
<td>-</td>
<td>0.0</td>
<td>x2</td>
<td>-</td>
</tr>
</tbody>
</table>

Notes:

- Two objective functions: the true objective (on top), and a fake one (below it).
- For *Phase I*, use the fake objective—it’s dual feasible.
- Two right-hand sides: the real one (on the left) and a fake (on the right).
- Ignore the fake right-hand side—we’ll use it in another algorithm later.
Network Simplex Tool

Data:
- **Costs** on arcs shown above arcs.
- **Supplies** at nodes shown above nodes.

Variables:
- **Primal flows** shown on tree arcs.
- **Dual slacks** shown on non-tree arcs.
- **Dual variables** shown below nodes.
Interior-Point Methods

Best illustrated on large-scale problems.

Click [here](#) to compare with the simplex method
Forthcoming Stuff

- *Integer Programming* (thanks to Hande Benson and her student).