Lurupa

Rigorous Error Bounds in Linear Programming

Christian Keil

Computer Science 3
Technical University Hamburg–Harburg

Algebraic and Numerical Algorithms
and Computer-assisted Proofs, 2005
Outline

1. Introduction

2. Numerical Experience

3. Software
Outline

1. Introduction
2. Numerical Experience
3. Software
A Representation of a Linear Program

Definition (Linear Program)

Find the optimal value $f^*$ of a linear objective function $c^T x$ subject to

- linear constraints $A x \leq a$, $B x = b$ and
- simple bounds $\underline{x} \leq x \leq \bar{x}$.

- Set of feasible points $F$ satisfying constraints and simple bounds
- Can be represented by the tuple $P := (c, A, a, B, b)$ and $\underline{x}, \bar{x}$
- Simple bounds may be infinite
A Lower Bound for the Optimal Value (1)

Theorem (Jansson, 2004)

Given a linear program $P$ and $x, \bar{x}$. If interval vectors $y \leq 0, z$ satisfy

- $\exists y \in \mathbf{y}, z \in \mathbf{z} : c_j - (A_{.j})^T y - (B_{.j})^T z = 0$ for free $x_j$, and
- the defects

$$d_j := c_j - (A_{.j})^T y - (B_{.j})^T z \begin{cases} \leq 0 & \text{for } -\infty < x_j \leq \bar{x}_j \\ \geq 0 & \text{for } x_j \leq x_j < \infty \end{cases}$$

then

$$f^* := \inf \{ a^T y + b^T z + \sum_{x_j \neq -\infty} x_j d^+_j + \sum_{\bar{x}_j \neq \infty} \bar{x}_j d^-_j \}$$

is a lower bound for the optimal value.
A Lower Bound for the Optimal Value (2)

- $O(n^2)$ operations for finite simple bounds, coincides with [Neumaier and Shcherbina, 2004]
- Conditions not met (infinite simple bounds) ⇒ iterate with perturbed dual constraints
- Works for interval problems $\mathbb{P}$
An Upper Bound for the Optimal Value

- Based on idea from [Krawczyk, 1975] later used and modified [Jansson, 1988], [Hansen and Walster, 1991], [Kearfott, 1994]
- Enclose primal interior point
- Verifies existence of primal feasible solutions in contrast to lower bound
Outline

1. Introduction

2. Numerical Experience

3. Software
A Real World Application Test Set

Netlib comprises ~ 100 problems

- First added in 1988, latest in 1996
- From e.g., stochastic forestry problems, oil refinery problems, flap settings of aircraft, pilot models, scheduling, truss structure, image restoration, industrial production and allocation models, multisector economic planning problems
- 32 to 15695 variables, 27 to 16675 constraints (medium size)
- Established test set for lp algorithms
- [Ordóñez and Freund, 2003]: 71% ill-conditioned
Condition and The Distance to Infeasibility

Definition (Distance to Infeasibility $\rho$)
Norm of the smallest perturbation of $P$ that results in an empty feasible set.

- $\rho_P$ distance to primal infeasibility
- $\rho_D$ distance to dual infeasibility

Definition (Condition)
Scale invariant reciprocal of the minimal distance to infeasibility.

$$C(d) := \frac{\|d\|}{\min\{\rho_P, \rho_D\}}$$
## Some Examples of What Can Happen

<table>
<thead>
<tr>
<th></th>
<th>( \rho_D )</th>
<th>( \rho_P )</th>
<th>( f^* )</th>
<th>( \bar{f}^* )</th>
<th>( \mu )</th>
</tr>
</thead>
<tbody>
<tr>
<td>scagr25</td>
<td>0.034646</td>
<td>0.021077</td>
<td>( \checkmark )</td>
<td>( \checkmark )</td>
<td>( 3.7821 \times 10^{-8} )</td>
</tr>
<tr>
<td>bnl1</td>
<td>0.106400</td>
<td>0</td>
<td>( \checkmark )</td>
<td></td>
<td>( 7.2244 \times 10^{-8} )</td>
</tr>
<tr>
<td>stair</td>
<td>0</td>
<td>0.000580</td>
<td></td>
<td>( \checkmark )</td>
<td>( 5.4796 \times 10^{-9} )</td>
</tr>
<tr>
<td>80bau3b</td>
<td>0</td>
<td>0</td>
<td>( \checkmark )</td>
<td>( \checkmark )</td>
<td>( 5.6653 \times 10^{-8} )</td>
</tr>
<tr>
<td>scsd8</td>
<td>1.000000</td>
<td>0.268363</td>
<td>( \checkmark )</td>
<td></td>
<td>( 6.3831 \times 10^{-8} )</td>
</tr>
</tbody>
</table>
Some Examples of What Can Happen

<table>
<thead>
<tr>
<th></th>
<th>$\rho_D$</th>
<th>$\rho_P$</th>
<th>$f^*$</th>
<th>$\bar{f}^*$</th>
<th>$\mu$</th>
</tr>
</thead>
<tbody>
<tr>
<td>scagr25</td>
<td>0.034646</td>
<td>0.021077</td>
<td>✓</td>
<td>✓</td>
<td>$3.7821 \times 10^{-8}$</td>
</tr>
<tr>
<td>bnl1</td>
<td>0.106400</td>
<td>0</td>
<td>✓</td>
<td>✓</td>
<td>$7.2244 \times 10^{-8}$</td>
</tr>
<tr>
<td>stair</td>
<td>0</td>
<td>0.000580</td>
<td></td>
<td>✓</td>
<td>$5.4796 \times 10^{-9}$</td>
</tr>
<tr>
<td>80bau3b</td>
<td>0</td>
<td>0</td>
<td>✓</td>
<td>✓</td>
<td>$5.6653 \times 10^{-8}$</td>
</tr>
<tr>
<td>scsd8</td>
<td>1.000000</td>
<td>0.268363</td>
<td>✓</td>
<td>✓</td>
<td>$6.3831 \times 10^{-8}$</td>
</tr>
</tbody>
</table>
## Dimensions of Example Problems

<table>
<thead>
<tr>
<th>Problem</th>
<th>m</th>
<th>p</th>
<th>n</th>
<th>nz</th>
</tr>
</thead>
<tbody>
<tr>
<td>scagr25</td>
<td>171</td>
<td>300</td>
<td>500</td>
<td>1554</td>
</tr>
<tr>
<td>bnl1</td>
<td>411</td>
<td>232</td>
<td>1175</td>
<td>5121</td>
</tr>
<tr>
<td>stair</td>
<td>147</td>
<td>209</td>
<td>467</td>
<td>3856</td>
</tr>
<tr>
<td>80bau3b</td>
<td>2262</td>
<td>0</td>
<td>9799</td>
<td>21002</td>
</tr>
<tr>
<td>scsd8</td>
<td>0</td>
<td>397</td>
<td>2750</td>
<td>8584</td>
</tr>
</tbody>
</table>
Overview of the Results

In total 89 problems, 86 with infinite simple bounds

- 35 finite upper bounds for the optimal value
  \[ \text{med}(\mu(\tilde{f}^*, f^*)) = 8.0e - 9 \quad \text{med}(t_{\tilde{f}^*}/t_{f^*}) = 5.3 \]

- 76 finite lower bounds for the optimal value
  \[ \text{med}(\mu(f^*, f^*)) = 2.2e - 8 \quad \text{med}(t_{f^*}/t_{f^*}) = 0.5 \]
Outline

1. Introduction

2. Numerical Experience

3. Software
Lurupa’s Design Goals

- Rigorous bounds for the optimal value
- Use unmodified solvers plus postprocessing
- Easy to use
- Easily adoptable to different solvers
- Standalone and library version
Lurupa’s Internals

- Implemented in ANSI C++
- Builds on interval library PROFIL/BIAS [Knüppel, 1994]
- Computational core
  uses solver modules to interface solvers
Usage of Lurupa

Example (command line)

```
>lurupa -sm /path/to/module \
   -lp /path/to/model -lb -ub
```

Meaning of parameters

- `sm` specify solver module
- `lp` specify lp model file
- `lb` compute lower bound
- `ub` compute upper bound
Usage of Lurupa

**Example (API)**

```c
Lurupa l;
l.set_solver_module("path/to/module");

FILE *in = fopen("path/to/model", "r");
Lp lp = l.read_lp(in, 0);

double optimal, bound, iterations;
l.solve_LP(lp, optimal);
l.lower_bound(lp, bound, iterations);
l.upper_bound(lp, bound, iterations);
```
Future Work

- Verified condition measures
- Sparse structures (in PROFIL)
- New solver modules
- Certificates of infeasibility, unboundedness
- Verified preprocessing [Fourer and Gay, 1993]
Outline
## Bounds and Distances to Infeasibility

<table>
<thead>
<tr>
<th></th>
<th>( \rho_P \geq 0 )</th>
<th>( \rho_P = 0 )</th>
<th>( \rho_D \geq 0 )</th>
<th>( \rho_D = 0 )</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>upper bound</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>finite</td>
<td>35</td>
<td>26</td>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>infinite</td>
<td>54</td>
<td>2</td>
<td>52</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>total</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>lower bound</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>finite</td>
<td>76</td>
<td>65</td>
<td>10</td>
<td>10</td>
<td>NA</td>
</tr>
<tr>
<td>infinite</td>
<td>13</td>
<td>3</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>total</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Christian Keil (TU Hamburg–Harburg)  
Lurupa  
Dagstuhl Seminar 05391
$\mu(\bar{f}^*, f^*)$ over $\rho_P$
$\mu(f^*, f^*)$ over $\rho_D$
## Overview of the Results

In total 89 problems, 86 with infinite simple bounds

<table>
<thead>
<tr>
<th></th>
<th>$\mu(_{\bar{f}^<em>}, f^</em>)$</th>
<th>$t_{\bar{f}^<em>}/t_{f^</em>}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>35 finite upper bounds</td>
<td>median 8.034e–9</td>
<td>5.250</td>
</tr>
<tr>
<td></td>
<td>mean 1.481e–2</td>
<td>38.776</td>
</tr>
<tr>
<td>76 finite lower bounds</td>
<td>median 2.183e–8</td>
<td>0.500</td>
</tr>
<tr>
<td></td>
<td>mean 7.279e–5</td>
<td>1.081</td>
</tr>
<tr>
<td>32 finite pairs</td>
<td>median 5.620e–8</td>
<td>5.000</td>
</tr>
<tr>
<td></td>
<td>mean 1.619e–2</td>
<td>42.317</td>
</tr>
</tbody>
</table>