Bad Honnef, November 18th 2010

Jan-Hendrik Jagla  
Alexander Meeraus

GAMS Development Corp.
GAMS Software GmbH
GAMS
Past Present Future
What’s that?

- Formulation mathematical optimization problems
- Notation similar to algebraic notation
- Ready-for-use links to state-of-the-art algorithms

► Simplified model building
► Efficient solution process
General Algebraic Modeling System

- Roots: World Bank, 1976
- Went commercial in 1987
- GAMS Development Corporation (Washington, Houston)
- GAMS Software GmbH (Köln, Braunschweig)

- Broad academic & commercial user community and network
(c) Regional farmer employment accounting rows:

\[ -RFS + 3 \sum_{w} \sum_{q} dFLq + \sum_{d} \sum_{r} dFL_t = 0, \text{ each } r \]

\[ \begin{bmatrix} \text{Regional farmer employment activity} \\ \text{Sum over districts and quarters of quarterly farmer employment} \end{bmatrix}^{37} + \begin{bmatrix} \text{Sum over districts and months of monthly farmer employment} \end{bmatrix}^{37} = 0 \]

(d) Total employment accounting row in man-years:

\[ -2LMAN + \sum_{r} LMAN_t = 0 \]

\[ -12 \begin{bmatrix} \text{Total employment in man-years} \end{bmatrix} + \begin{bmatrix} \text{Sum over months of total employment in man-months} \end{bmatrix} = 0 \]

(e) Total monthly employment accounting rows in man-months:

\[ -2.2LMAN_t + \sum_{d} dDL_t + \sum_{d} dFLq + \sum_{d} dFL_t = 0, \text{ each } t \text{ and } q \text{ such that } t \in q \]

\[ -2.2 \begin{bmatrix} \text{Total employment in month } t \end{bmatrix} + \begin{bmatrix} \text{Sum over districts of day labor employment in month } t \end{bmatrix}^{38} + \begin{bmatrix} \text{Sum over districts of quarterly farmer employment in the quarter containing month } t \end{bmatrix} + \begin{bmatrix} \text{Sum over districts of monthly farmer employment} \end{bmatrix}^{38} = 0 \]

\(^{37}\) In irrigation districts the quarterly contract device is used for farmers, but in non-irrigated districts farmers are assumed to be available on a monthly basis, so that seasonal migration to irrigated areas may occur.

\(^{38}\) The activities for hiring farmers and day laborers are stated in units of tens of man-days per month (or quarter), and there are 22 working days per month; hence the conversion factor of 2.2 is required in the first term of this equation.
# Model Data

## Table 3

Sequence of standard operations for cotton cultivation (days of unskilled labor, machinery services, and draft animal services required per hectare by month)

<table>
<thead>
<tr>
<th>Cultivation month and operation</th>
<th>Mechanized</th>
<th>Partially mechanized</th>
<th>Non-mechanized</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Unskilled labor</td>
<td>Machinery</td>
<td>Unskilled labor</td>
</tr>
<tr>
<td>1st Preparatory tasks</td>
<td>0.12</td>
<td></td>
<td>0.12</td>
</tr>
<tr>
<td>Fallow</td>
<td>0.5</td>
<td></td>
<td>0.5</td>
</tr>
<tr>
<td>Cross-plowing</td>
<td>0.2</td>
<td></td>
<td>0.2</td>
</tr>
<tr>
<td>Harrowing</td>
<td>0.25</td>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td>Land levelling</td>
<td>1.0</td>
<td></td>
<td>1.0</td>
</tr>
<tr>
<td>Canal cleaning</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2nd Irrigation ditches</td>
<td>1.0</td>
<td>0.2</td>
<td>1.0</td>
</tr>
<tr>
<td>Forming borders</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Linking borders</td>
<td>1.0</td>
<td>0.2</td>
<td>1.0</td>
</tr>
<tr>
<td>Water application</td>
<td>2.0</td>
<td>0.2</td>
<td>2.0</td>
</tr>
<tr>
<td>Harrowing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Seeding and fertilization</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td>Maintenance of field works</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3rd Thinning plants</td>
<td>4.0</td>
<td>0.2</td>
<td>4.0</td>
</tr>
<tr>
<td>Cultivation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weeding</td>
<td>6.0</td>
<td></td>
<td>6.0</td>
</tr>
<tr>
<td>Applications of pesticides (2)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Matrix Generator Input

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>AGGREGAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>AZU</td>
<td>AZU</td>
<td>-0.25</td>
<td>1.0</td>
<td>0.0070</td>
<td>2627020.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>JIT</td>
<td>0.4</td>
<td>1.0</td>
<td>0.1150</td>
<td>174752.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PEP</td>
<td>PEP</td>
<td>0.6</td>
<td>1.0</td>
<td>0.0560</td>
<td>10.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PLU</td>
<td>PLU</td>
<td>-1800.0</td>
<td>1.0</td>
<td>0.5770</td>
<td>85209.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CHI</td>
<td></td>
<td>1</td>
<td>-0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CHV</td>
<td></td>
<td>0.1500</td>
<td>14459</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FDR</td>
<td></td>
<td>0</td>
<td>-0.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRD</td>
<td></td>
<td>0.0630</td>
<td>205818.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CEG</td>
<td></td>
<td>0.0930</td>
<td>0.865</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALV</td>
<td></td>
<td>0.0100</td>
<td>226109.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALA</td>
<td></td>
<td>0.0400</td>
<td>179019.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAR</td>
<td></td>
<td>0.0990</td>
<td>17427.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAI</td>
<td></td>
<td>0.0860</td>
<td>77997.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FEC</td>
<td></td>
<td>4</td>
<td>-0.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FRI</td>
<td></td>
<td>0.1830</td>
<td>33001.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARQ</td>
<td></td>
<td>0.1220</td>
<td>176197.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PAP</td>
<td></td>
<td>0.0930</td>
<td>27138.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GAR</td>
<td></td>
<td>0.0990</td>
<td>17427.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GRA</td>
<td></td>
<td>2</td>
<td>-0.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEL</td>
<td></td>
<td>0.0860</td>
<td>142804.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRI</td>
<td></td>
<td>0.0800</td>
<td>343979.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FRU</td>
<td></td>
<td>2</td>
<td>-2.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAL</td>
<td></td>
<td>0.0780</td>
<td>10850.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEL</td>
<td></td>
<td>0.0930</td>
<td>69350.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAL</td>
<td></td>
<td>0.0830</td>
<td>193910.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JON</td>
<td></td>
<td>0.2410</td>
<td>97224.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CAR</td>
<td></td>
<td>0.1550</td>
<td>77490.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SOY</td>
<td></td>
<td>0.1600</td>
<td>57220.0</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>END</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALA</td>
<td></td>
<td>0.02</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALV</td>
<td></td>
<td>-1</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Variable</td>
<td>Constraint</td>
<td>Value</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>------------</td>
<td>-------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C1</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C2</td>
<td>0.99400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C3</td>
<td>0.33500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C4</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C5</td>
<td>0.05000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C6</td>
<td>0.15000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C7</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C8</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C9</td>
<td>0.20600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C10</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C11</td>
<td>0.98400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C12</td>
<td>0.15000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C13</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C14</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C15</td>
<td>0.98400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C16</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C17</td>
<td>0.98400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C18</td>
<td>0.98400</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C19</td>
<td>0.20600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C20</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C21</td>
<td>7.56000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C22</td>
<td>0.31000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C23</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C24</td>
<td>7.56000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C25</td>
<td>0.33500</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C26</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C27</td>
<td>7.56000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C28</td>
<td>0.20600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C29</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C30</td>
<td>0.05000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C31</td>
<td>0.15000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C32</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C33</td>
<td>7.56000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C34</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C35</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C36</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C37</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C38</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C39</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td>C40</td>
<td>1.00000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BRANCH</td>
<td>NEXT</td>
<td>REVISE</td>
<td>REVS</td>
<td>MAJERR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
<td>--------</td>
<td>------</td>
<td>--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HEADERS, CARD NO.</th>
<th>ENAME</th>
<th>REVA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>@COLUMNS</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>@MODIFY</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>@RHS</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>@MODIFY</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>@ENDATA</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>@NAME</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>@COLUMNS</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>@MODIFY</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>@ENDATA</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>@NAME</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>@COLUMNS</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>@MODIFY</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>@ENDATA</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>@NAME</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>@RHS</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>@MODIFY</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>@ENDATA</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>@NAME</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>@RHS</td>
<td></td>
</tr>
<tr>
<td>71</td>
<td>@MODIFY</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>@RHS1</td>
<td>CLAV_01 5.03328</td>
</tr>
<tr>
<td>73</td>
<td>@RHS1</td>
<td>CLAV_02 5.03328</td>
</tr>
<tr>
<td>74</td>
<td>@RHS1</td>
<td>CLAV_03 5.03328</td>
</tr>
<tr>
<td>75</td>
<td>@RHS1</td>
<td>CLAV_04 5.03328</td>
</tr>
<tr>
<td>76</td>
<td>@RHS1</td>
<td>CLAV_05 5.03328</td>
</tr>
<tr>
<td>77</td>
<td>@RHS1</td>
<td>CLAV_06 5.03328</td>
</tr>
<tr>
<td>78</td>
<td>@RHS1</td>
<td>CLAV_07 5.03328</td>
</tr>
<tr>
<td>79</td>
<td>@RHS1</td>
<td>CLAV_08 5.03328</td>
</tr>
<tr>
<td>80</td>
<td>@RHS1</td>
<td>CLAV_09 5.03328</td>
</tr>
<tr>
<td>81</td>
<td>@RHS1</td>
<td>CLAV_10 5.03328</td>
</tr>
<tr>
<td>82</td>
<td>@RHS1</td>
<td>CLAV_11 5.03328</td>
</tr>
<tr>
<td>83</td>
<td>@RHS1</td>
<td>CLAV_12 5.03328</td>
</tr>
<tr>
<td>84</td>
<td>@RHS1</td>
<td>CLAV_13 5.03328</td>
</tr>
<tr>
<td>85</td>
<td>@ENDATA</td>
<td></td>
</tr>
</tbody>
</table>
### MPS Output

**DATE**: 07/30/76  
**TIME**: 22:12:21  

<table>
<thead>
<tr>
<th>NUMBER</th>
<th>NAME</th>
<th>TYPE</th>
<th>STATUS</th>
<th>COL ACTIVITY</th>
<th>OBJ COEF</th>
<th>D_UPPER</th>
<th>MARGINAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>CBE1C</td>
<td>PL</td>
<td>LOWER</td>
<td></td>
<td></td>
<td>INF</td>
<td>-8,44851</td>
</tr>
<tr>
<td>102</td>
<td>CBE2F</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-87,80000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>103</td>
<td>CBE3C</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-801,0000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>104</td>
<td>CBE4F</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-91330,6000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>105</td>
<td>CBE5C</td>
<td>PL</td>
<td>LOWER</td>
<td>.00025</td>
<td>-2424,7000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>106</td>
<td>CBE6C</td>
<td>PL</td>
<td>LOWER</td>
<td>.00025</td>
<td>-9446,0000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>107</td>
<td>CBE7C</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-5116,0000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>108</td>
<td>CSG.V</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-13,20000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>109</td>
<td>CSG.F</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-231,57000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>110</td>
<td>CPO.V</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-139,87000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>111</td>
<td>CPO.F</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-134,67000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>112</td>
<td>CPO.C</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-78,71000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>113</td>
<td>CEG.V</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-78,71000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>114</td>
<td>CEG.F</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-76,71000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>115</td>
<td>CEG.C</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-76,71000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>116</td>
<td>CDA.C</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>-76,71000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>117</td>
<td>COF.C</td>
<td>PL</td>
<td>LOWER</td>
<td>.00025</td>
<td>-129,91000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>118</td>
<td>COD.C</td>
<td>PL</td>
<td>LOWER</td>
<td>.00025</td>
<td>-180,74000</td>
<td>INF</td>
<td>-87,19134</td>
</tr>
<tr>
<td>119</td>
<td>COS.C</td>
<td>PL</td>
<td>LOWER</td>
<td>.00025</td>
<td>-187,65000</td>
<td>INF</td>
<td>-256,3963</td>
</tr>
<tr>
<td>120</td>
<td>COL.C</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>107,65000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>121</td>
<td>CHS.C</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>121,35000</td>
<td>INF</td>
<td></td>
</tr>
<tr>
<td>122</td>
<td>CHL.C</td>
<td>PL</td>
<td>ACTIVE</td>
<td>.00025</td>
<td>91,60000</td>
<td>INF</td>
<td></td>
</tr>
</tbody>
</table>
PLANNING PROBLEM AND OBJECTIVES INITIALLY OFTEN

UNSTRUCTURED
ILL-DEFINED
CONFLICTING
UNCERTAIN
CHANGING
EMOTIONAL

MATHEMATICAL MODEL USED TO RECOGNIZE AND FORMULATE PROBLEMS, DEFINE ISSUES AND EXPLORE SOLUTION SPACE
PRESENT TECHNOLOGY

REAL WORLD PROBLEM

SOLUTION

ANALYST

MODEL

DATA

Operating Systems
Model Generators
Computer Languages
Report Generators
Data Base Systems
Solution Packages

RESULT: - Drain of resources (technical, time, money)
- Essentially no documentation
MAJOR CONSTRAINTS: COST

SKILLS

TIME

TOOLS

DOCUMENTATION

TRUST
RESULT:
- Limited drain of resources
- Same representation of models for humans and machines
- Model representation is also model documentation
DEVELOPMENT OF GAMS

Phase 1 (1978)

- The system can be used to represent and analyze any algebraic model (be it linear or nonlinear)
- The system can perform algebraic manipulations on all data
- The system can generate and solve linear programs automatically
- The system can generate reports on data and solutions via simple 'display' statements
DEVELOPMENT OF GAMS

Phase 2 (1979)

- The system can generate and solve nonlinear programs

- The system will provide links to special-purpose algorithms for econometric problems, network problems, etc.

- Appropriate extensions to the language will be made as the need arises
DEVELOPMENT OF GAMS

Phase 3 (?)

- Automatic structure recognition

- Internal generation of exact point-derivatives

- Improved data-base design with e.g. unit analysis, and links to existing data bases

- Availability of GAMS on different machines

- World-wide availability of the system so that it can be used as a market for testing models and algorithms
GAMS’ Fundamental concepts

- Platform independence
- Hassle-free switch of solution methods
- Open architecture and interfaces to other systems
- Balanced mix of declarative and procedural elements

10+ Supported Platforms

- Solaris 64bit
- Solaris
- AXU
- AIX
- Linux 64bit
- Mac
- Windows
- Linux
- Window 64bit
- HP
GAMS’ Fundamental concepts

- Platform independence
- Hassle-free switch of solution methods
- Open architecture and interfaces to other systems
- Balanced mix of declarative and procedural elements
GAMS’ Fundamental concepts

- Platform independence
- Hassle-free switch of solution methods
- Open architecture and interfaces to other systems
- Balanced mix of declarative and procedural elements

Gams Data eXchange (GDX)

API’s
- Gams Data eXchange (GDX)
- Options
- Gams Modeling Object (GMO)
- Gams Environment (GEV)
- ...

Layers of separation
GAMS’ Fundamental concepts

• Layers with separation of
  – model and data
  – model and solution methods
  – model and operating system
  – model and interface

➔ Models benefit from
  – advancing hardware
  – enhanced / new solver technology
  – improved / upcoming interfaces to other systems
GAMS’ Fundamental concepts

- Platform independence
- Hassle-free switch of solution methods
- Open architecture and interfaces to other systems
- Balanced mix of declarative and procedural elements

Declarative Elements
- Sets
- Parameters
- Variables
- Equations
- Models
- ...

Procedural Elements
- loops
- if-then-else
- ...

GAMS Fundamental concepts
Problem 1

• Data transfer between different systems slow, error prone and bulky.

• Application (real time) require the capture of data instances that can be analyzed off-line in other environments.

• Management of name space mappings between different problems and their transformations into other data representations.

• Separate the model from its environment

• Search for a common low level high performance data container
Problem 2

- Building and maintaining solver specific links in different programming languages became a huge resource sink and made the introduction of new features difficult.
  - Simplify the building and maintaining of solver links
  - Manage multiple interacting models
  - Minimize the solver specific tailoring
  - Maintain one source only
  - Wrap automatically for different languages.

- Share libraries between the data management part of a modeling system and the solver. Example: function evaluations, first and second order derivatives, intervals,..

- Ease linking of experimental (meta-)solvers to GAMS
Problem 3

• Problems may contain
  – Complementarity
  – Hierarchy
  – Interacting agents
  – Risk measures
  – Logic relationships
• Cannot be expressed with current modeling languages and have no direct solution method.
• Example: General equilibrium models are a transformation from multi agent optimization/variational problems into a single mixed complementarity model.
• How to automate the transformations by annotations of existing optimization models that convey model structure to the solver.
GAMS
Features you might not know
Problem 1

• Data transfer between different systems slow, error prone and bulky.

• Application (real time) require the capture of data instances that can be analyzed off-line in other environments.

• Management of name space mappings between different problems and their transformations into other data representations.

• Separate the model from its environment

• Search for a common low level high performance data container
GDX
Gams Data eXchange
Gams Data eXchange

Binary Data Exchange

- Fast
- Safes disk space
- Tailored for large sparse matrices
- Platform independent
- Direct GDX interfaces
- API support for high-level programming languages
- Utilities

GDX Utilities

- Invert
- IDE
- GDX Viewer
- GDXrank
- GDX2HAR / HAR2GDX
- GDXmerge
- GDXdump
- GDXcopy
- MDB2GMS
- GDXdiff
- GDXxrw
- GDX2XLS
- GAMS
- GDX API
GAMS in Control
Application in Control

Creating Input

GDX Container

Call GAMS

GAMS (Executable / DLL)

GDX Container

Reading Solution

GDX API

GDX API
Incorporating a model in a spreadsheet

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
<th>M</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Distance</td>
<td>New-York</td>
<td>Chicago</td>
<td>Topeka</td>
<td>Supply</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Seattle</td>
<td>2.5</td>
<td>1.7</td>
<td>1.8</td>
<td>350</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>San-Diego</td>
<td>2.5</td>
<td>1.8</td>
<td>1.4</td>
<td>600</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Demand</td>
<td>325</td>
<td>300</td>
<td>275</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Freight cost</td>
<td>90</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>SHIPMENT</td>
<td>New-York</td>
<td>Chicago</td>
<td>Topeka</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Seattle</td>
<td>50</td>
<td>300</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>San-Diego</td>
<td>275</td>
<td>0</td>
<td>275</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Solver: CPLEX
Equations: 6 Variables: 7
Model Status: 1 Optimal
Solver Status: 1 Normal Completion
Iterations: 4 Solve Time: 0.00
Objective Value: 153.675
Problem 2

• Building and maintaining solver specific links in different programming languages became a huge resource sink and made the introduction of new features difficult.

  → Simplify the building and maintaining of solver links
  → Manage multiple interacting models
  → Minimize the solver specific tailoring
  → Maintain one source only
  → Wrap automatically for different languages.

• Share libraries between the data management part of a modeling system and the solver. Example: function evaluations, first and second order derivatives, intervals,..

• Ease linking of experimental (meta-)solvers to GAMS
GMO

Gams Modeling Object
Gams Modeling Object (GMO)

GAMS’ Next-Generation Model API

- Why a new model API?
- What do we need it to do?
- What does it look like? How is it put together?
- How did we do it?
- When are we going to be finished?
Solver Links – Different Perspectives

GAMS User
• Standardized solver interface allows “hassle free” replacement of solvers: `option nlp=conopt;`  
  …nothing will change

Solver & Solver-link Developer
• IO Library provides access to
  – Matrix
  – Function/Gradient/Hessian evaluations
  – Solution file writer
  – Output handling
  – GAMS Options (e.g. resource limit)
  – Problem attributes (SOS, semicont, semiint, priorities, scales)
  – Utility routines
  – problem rewriting, matrix reordering
  …our focus here
Reuse? What’s that?!?

Solver Links

- AlphaECP
- Baron
- Xa
- Xpress (~50 total)

- Conopt
- Minos
- Snopt
- ... PATH
- ... MILES
- ... Cplex
- GUROBI
- XPRESS

- Fortran I/O Library
- Delphi I/O Library
- C I/O Library
- MPEC I/O
- SMAG Lib
- G2D Lib
Advantages

• proven over many years
• all platforms supported
• all GAMS-features available
• written by language experts, use all language features
• resulted in high-quality links across solvers and platforms
  → has been one factor in our success
Disadvantages

• Not always intuitive to use

  Linking your Solver to GAMS - THE COMPLETE NOTES (160 pages !!)

• Outdated design – I/O, STOP

• feature-poor (e.g. no automatic reformulation of objective func/var)

• inconvenient & expensive to maintain

• painful to move ‘inert mass’ forward

• linking your solver (without buddy at GAMS) is very difficult
Philosophy behind GMO

• Then
  – Computing environment: limited time and memory
  – Algorithm APIs not uniform or language-neutral
  – Expert users who understand optimization
  – Don’t use unnecessary space or time
  – If the link gets in trouble, just abort, the user will fix things up and re-run.

• Now
  – Most users won’t hit space/time limits
  – APIs look similar, are language-neutral
  – Users may be domain experts, not MP experts
  – Use of additional space & time to give the GMO and GAMS user a better experience is justified.
Checklist for GMO

- Powerful & convenient API – a few calls do the job
- In-core communication between GAMS and the solver, making potentially large model scratch files unnecessary
- Implement once, run everywhere (multiple platforms & multiple languages)
  - Platform-independent code, isolate the “dirty bits”.
  - API wrapper & multi-language interface
- Support meta-solvers (e.g. DICOPT, SBB, Examiner)
- Separate Model from Environment
- Comprehensive – one-stop shop for all linking needs
- Support shared-library implementation of solver links
- Support multiple models
Checklist for GMO

- Powerful & convenient API – a few calls do the job
- In-core communication between GAMS and the solver, making potentially large model scratch files unnecessary
- Implement once, run everywhere (multiple platforms & multiple languages)
  - Platform-independent code, isolate the “dirty bits”.
  - API wrapper & multi-language interface
- Support meta-solvers (e.g. DICOPT, SBB, Examiner)
- Separate Model from Environment
- Comprehensive – one-stop shop for all linking needs
- Support shared-library implementation of solver links
- Support multiple models
GMO: Powerful and Convenient API

• What’s a powerful call?
  – Basic CS: information hiding, encapsulation, object model, abstraction
  – One call to do the job required, e.g. Hessian setup
  – No preconditions, magic calls, or nasty side effects

• Convenience - multiple routines and “flavors”
  – Jacobian row- vs. column-wise, tuples
  – Objective reformulation – function or variable
  – Free rows - yes or no
  – Column evals: dense or sparse, all or just NL
  – Common/typical tasks done in GMO, not the link
Checklist for GMO

- Powerful & convenient API – a few calls do the job
- In-core communication between GAMS and the solver, making potentially large model scratch files unnecessary
- Implement once, run everywhere (multiple platforms & multiple languages)
  - Platform-independent code, isolate the “dirty bits”.
  - API wrapper & multi-language interface
- Support meta-solvers (e.g. DICOPT, SBB, Examiner)
- Separate Model from Environment
- Comprehensive – one-stop shop for all linking needs
- Support shared-library implementation of solver links
- Support multiple models
solve mymodel minimizing z using lp
mymodel.solvelink = {ChainScript, CallScript, 
    CallModule, AsyncGrid, AsyncSimulate, LoadLibrary};

• ChainScript: Solver process, GAMS vacates memory
  + Maximum memory available to solver
  + protection against solver failure (*hostile* link)
  - swap to disk

• Call{Script/Module}: Solver process, GAMS stays live
  + protection against solver failure (*hostile* link)
  + no swap of GAMS database
  - file based model communication
Solver Integration – cont.

- LoadLibrary: Solver DLL in GAMS process
  + fast memory based model communication
  - not (yet) supported by all solvers

- transport.gms (LP) solved 500 times with CPLEX:

```gams
set ss /s1*s500/;
loop {ss,
solve transport minimizing z using lp;
}
```

- ChainScript: 33.04 s (28.9s)*
- CallModule: 13.78 s (12.7s)
- LoadLibrary: 2.37 s (2.0s)

* without Virus Scanner
Checklist for GMO

- Powerful & convenient API – a few calls do the job
- In-core communication between GAMS and the solver, making potentially large model scratch files unnecessary
- Implement once, run everywhere (multiple platforms & multiple languages)
  - Platform-independent code, isolate the “dirty bits”.
  - API wrapper & multi-language interface
- Support meta-solvers (e.g. DICOPT, SBB, Examiner)
- Separate Model from Environment
- Comprehensive – one-stop shop for all linking needs
- Support shared-library implementation of solver links
- Support multiple models
Implement Once, Run Everywhere

• All GMO coding done in a single language and style
  – Allows code sharing with other components
  – Allows for shared development (GMO is a team effort)

• All GMO coding is platform-independent
  – Makes writing code faster, more reliable
  – Maintenance is simplified

• Platform-dependent code isolated in utility libraries
  – Makes adding a new platform easier
  – Maintenance is simplified
  – Unit testing is easy and effective
Automated Generation of APIs

‘The GAMS Wrapper’

- API is defined using the GAMS language
- A tool written in GAMS is used to regenerate APIs for all languages
- Executed on request and nightly

→ A change in the definition of the API immediately makes it into all language interfaces
→ No manual and therefore error-prone efforts required
Automated Generation of APIs

‘The GAMS Wrapper’

- Automated nightly testing

- API version checks

- Reusable for multiple GAMS component libraries
  - GMO
  - GAMS
  - GDX
  - Option
Distributed GAMS APIs

- Component Libraries
  - GAMS
  - GDX
  - Option

- Supported languages
  - C, C++, C#
  - Delphi
  - Fortran
  - Java
  - VBA, VB.Net
  - Python

- Examples/Documentation
Checklist for GMO

- Powerful & convenient API – a few calls do the job
- In-core communication between GAMS and the solver, making potentially large model scratch files unnecessary
- Implement once, run everywhere (multiple platforms & multiple languages)
  - Platform-independent code, isolate the “dirty bits”.
  - API wrapper & multi-language interface
- Support meta-solvers (e.g. DICOPT, SBB, Examiner)
- Separate Model from Environment
- Comprehensive – one-stop shop for all linking needs
- Support shared-library implementation of solver links
- Support multiple models
Meta-Solvers with GMO

- Populated GMO object (e.g. by GAMS)
- GMO API to allow modification and alteration of bounds, rhs, “modifiable” parameters (NL expression evaluation)
- GMO/GEV (GAMS Environment Object) based solver links
- Runtime system (C, Python, Java, …)

- Alternative way to implement decomposition, and other algorithmic ideas based on MP models
‘Efficient’ Implementation of B&B

---

GAMS Process
Model Files
Solution Files
Solver Process
SBB Process
Bound Revision File
Node File Primal + Dual Solution
RMINLP Solution Files
NLP Solver Process

---

SBB, 2001
‘Efficient’ Implementation of B&B

GAMS Process

Model Files

Solution Files

Solver Process

SBB Process

Bound Revision File

Node File Primal + Dual Solution

RMINLP Solution Files

NLP Solver Process

SBB, 2001
‘Efficient’ Implementation of B&B

GAMS Process

Model Files

Solution Files

Solver Process

SBB Process

Bound Revision File

Node File Primal + Dual Solution

RMINLP Solution Files

NLP Solver Process

SBB, 2001
Dicopt (Outer Approximation)

If MASTER infeasible STOP

If Upper Bound is $\varepsilon$ larger than Lower Bound
STOP

Add linearizations from solution of NLP ($y_j$)

Fix $y_j$

MASTER (MIP)

Lower Bound

Upper Bound

NLP($y_j$)
Series of NLP and MIP solves

--- DICOPT: Log File:

<table>
<thead>
<tr>
<th>Step</th>
<th>Major</th>
<th>Objective</th>
<th>CPU time</th>
<th>Iterations</th>
<th>Evaluation Errors</th>
<th>Solver</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLP 1</td>
<td>1</td>
<td>1.04923</td>
<td>0.02</td>
<td>38</td>
<td>0</td>
<td>conopt</td>
</tr>
<tr>
<td>MIP 1</td>
<td>1</td>
<td>9.07274</td>
<td>0.09</td>
<td>28</td>
<td>0</td>
<td>cplex</td>
</tr>
<tr>
<td>NLP 2</td>
<td>2</td>
<td><em>Infeas</em></td>
<td>0.00</td>
<td>10</td>
<td>0</td>
<td>conopt</td>
</tr>
<tr>
<td>MIP 2</td>
<td>2</td>
<td>13.02091</td>
<td>0.13</td>
<td>32</td>
<td>0</td>
<td>cplex</td>
</tr>
<tr>
<td>NLP 3</td>
<td>3</td>
<td>1.26864&lt;</td>
<td>0.03</td>
<td>27</td>
<td>0</td>
<td>conopt</td>
</tr>
<tr>
<td>MIP 3</td>
<td>3</td>
<td>13.93760</td>
<td>0.11</td>
<td>29</td>
<td>0</td>
<td>cplex</td>
</tr>
<tr>
<td>NLP 4</td>
<td>4</td>
<td><em>Infeas</em></td>
<td>0.02</td>
<td>7</td>
<td>0</td>
<td>conopt</td>
</tr>
<tr>
<td>MIP 4</td>
<td>4</td>
<td>13.99258</td>
<td>0.11</td>
<td>19</td>
<td>0</td>
<td>cplex</td>
</tr>
<tr>
<td>NLP 5</td>
<td>5</td>
<td><em>Infeas</em></td>
<td>0.02</td>
<td>13</td>
<td>0</td>
<td>conopt</td>
</tr>
<tr>
<td>MIP 5</td>
<td>5</td>
<td>21.03812</td>
<td>0.11</td>
<td>23</td>
<td>0</td>
<td>cplex</td>
</tr>
<tr>
<td>NLP 6</td>
<td>6</td>
<td>1.26864</td>
<td>0.02</td>
<td>17</td>
<td>0</td>
<td>conopt</td>
</tr>
</tbody>
</table>

--- DICOPT: Terminating...

• Lots of file writing and reading to communicate between Dicopt, MIP, and NLP solver

• Basically start a whole new process over and over
New Dicopt Implementation

Joined work with Ignacio Grossmann, Juan Pablo Ruiz (Carnegie Mellon University)

- Object Oriented

- Use C++ Interface to GMO

- Two models: NLP and MIP

- Use standardized solver interface to call NLP/MIP solver in-core (pass GMO ‘handle’ on to solver)

- Algorithmic improvements
• Everybody agrees: Version control is a game-changer
### Latest GAMS System Builds and Test Results

<table>
<thead>
<tr>
<th>System</th>
<th>Libraries</th>
<th>Build</th>
<th>Rev</th>
<th>Status and Time (UTC)</th>
<th>Initial Tests</th>
<th>Full Tests</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inx</td>
<td>Download</td>
<td>23.7.0</td>
<td>21065</td>
<td>Test done</td>
<td>712 runs 0 failures (q=0,s=0)</td>
<td>Report 8902 runs 2 failures (q=1,s=1)</td>
</tr>
<tr>
<td>Inx</td>
<td>Download</td>
<td>23.7.0</td>
<td>21065</td>
<td>Test done</td>
<td>732 runs 0 failures (q=0,s=0)</td>
<td>Report 9385 runs 2 failures (q=1,s=1)</td>
</tr>
<tr>
<td>vs8</td>
<td>Download</td>
<td>23.7.0</td>
<td>21075</td>
<td>Test done</td>
<td>734 runs 0 failures (q=0,s=0)</td>
<td>Report 9397 runs 2 failures (q=1,s=1)</td>
</tr>
<tr>
<td>weih</td>
<td>Download</td>
<td>23.7.0</td>
<td>21072</td>
<td>Test done</td>
<td>682 runs 1 failures (q=1,s=0)</td>
<td>Report 8364 runs 5 failures (q=2,s=3)</td>
</tr>
<tr>
<td>Inx</td>
<td>Download</td>
<td>23.6.0</td>
<td>21070</td>
<td>Test done</td>
<td>712 runs 0 failures (q=0,s=0)</td>
<td>Report 8901 runs 2 failures (q=1,s=1)</td>
</tr>
</tbody>
</table>
How We Did It: Automation & Testing

- SVN and other tools automate builds on all platforms

- Extensive, automated tests
  - Test library (503 models), other libraries (hundreds)
  - Runs over all solvers, some NLP/MIP combinations
  - Recent beta: 17 test machines, each ~ 3K – 10K
  - Collecting, archiving, sharing of test results

- PAVER used to compare to previous versions
  - Helps find outliers (bugs), problem cases
When Will We Be Finished?

- **GAMS 23.5.2 (current distribution)**
  - Coin-Solvers (Bonmin, Cbc, Couenne, Ipopt, OS)
  - Gurobi
  - Lindoglobal
  - OSI-based links to Cplex, Gurobi, Glpk, Mosek, Xpress
  - Scip
  - …

- **GAMS 23.6 (currently in beta)**
  - All previous Fortran links (e.g. Conopt, Minos, Snopt)
  - All links using 2\textsuperscript{nd}-order information (e.g. Knitro, Pathnlp, Mosek)
  - Xpress

- **GAMS 23.7 ??**
Summary

Solver Links

AlphaECP
Baron
......
Xa
Xpress
(~50 total)

Fortran I/O
Library

Delphi I/O
Library

C I/O
Library

MPEC I/O

SMAG Lib

G2D Lib
Summary

 Solver Links

 AlphaECP
 Baron
 ...... 
 Xa
 Xpress
 (~50 total)

 Your solver!

 Gams
 Modeling
 Object

 CONOPT
 MINOS
 SNOPT
 ...
 PATH
 MILES
 ...
 CPLEX
 GUROBI
 XPRESS

 Your solver!
Summary

• GMO is part of GAMS distribution

• GMO is used by a variety of / will be used by all GAMS Solver Links

• GMO eases maintenance and makes development process more flexible, more agile

• GMO opens up new possibilities for moving GAMS forward

• GMO interfaces are not yet public but alpha version can be made available on request
Scenario Solver

Use GDX and GMO
Loop(s,
    d(i,j) = dd(s,i,j);
    f = ff(s);
    solve mymodel min z using lp;
    rep(s) = mymodel.objval;
);

<table>
<thead>
<tr>
<th>Setting</th>
<th>Solve time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solvelink=0 (default)</td>
<td>40.297</td>
</tr>
<tr>
<td>Solvelink=%Solvelink.LoadLibrary%</td>
<td>03.625</td>
</tr>
</tbody>
</table>
GAMS Scenario Solver

cost.. 

z=e=sum((i,j),f*d(i,j)/1000*x(i,j));

set dict / s.scenario."

  d.param .dd
  f.param .ff
  x.level .xx /

solve mymodel min z using lp scenario dict;

• Update model data instead of matrix coefficients/rhs
• Hot start (keep the model hot inside the solver and use solver’s best update mechanism)
• Save model generation and solver setup time
• Model rim unchanged from scenario to scenario
• Apriori knowledge of all scenario data
Problem 3

- Problems may contain
  - Complementarity
  - Hierarchy
  - Interacting agents
  - Risk measures
  - Logic relationships

- Cannot be expressed with current modeling languages and have no direct solution method.

- Example: General equilibrium models are a transformation from multi agent optimization/variational problems into a single mixed complementarity model.

- How to automate the transformations by annotations of existing optimization models that convey model structure to the solver.
EMP
Extended Math. Programming
Current state: Model-Side

- Traditional problem format

\[
\begin{align*}
\min_x c(x) \quad & \text{s.t. } A_1(x) \leq b_1, \ A_2(x) = b_2
\end{align*}
\]

- Support for complementarity constraints

- Interactions between models possible
  - Series of models
  - Scenario analyses / parallelized model runs
  - Iterative sequential feedback
  - Decomposition
New solution concepts

- Embedded Complementarity Systems
- Disjunctive Programs
- Bilevel Programs
- Extended Nonlinear Programs
- Variational Inequalities
- …

- Breakouts of traditional MP classes
- No conventional syntax
- Limited support with common model representation
- Incomplete/experimental solution approaches
- Lack of reliable/any software
Do not:
• overload existing GAMS notation right away!
• attempt to build new solvers right away!

But:
• Use existing language features to specify additional model features
• Distribute information as part of the production system
• Express extended model in symbolic form and apply existing matured solution technology

⇒ Extended Mathematical Programming (EMP)
GAMS “Solver” EMP

EMP Information

Original Model

Translation

Reformulated Model

Viewable

Solving using established Algorithms

Solution

Mapping Solution into original space
Bilevel Programming

\[
\begin{align*}
\min_{x,y} & \quad f(x, y) \\
\text{s.t.} & \quad g(x, y) \leq 0, \\
& \quad y \text{ solves } \min_{s} v(x, s) \text{ s.t. } h(x, s) \leq 0
\end{align*}
\]

• Additional Information:

$\texttt{onecho > %emp.info%}$

Bilevel x min v h

$\texttt{offecho}$

• EMP Tool automatically creates an MPEC by expressing the lower level optimization problem through its optimality conditions
Bilevel Model


defobj..  z  =e=  sqr(x1+x2-2) + sqr(x3+x4-2);
a0..  h1  =e=  h1_0;
a1..  h2  =e=  h2_0;
a2..  x1-x2  =e=  3;

Outer Problem

defh1..  h1  =e=  sqr(u1-x1) + sqr(u2-x2) + sqr(u3-x3) + sqr(u4-x4);
e1..  3*u1 + u2 + 2*u3 + u4  =e=  6;

Inner Problem 1

defh2..  h2  =e=  sqr(v1-x1) + sqr(v2-x2) + sqr(v3-x3) + sqr(v4-x4);
e2..  v1 + v2 + v3 + 2*v4  =e=  7;

Inner Problem 2
EMP Information File + EMP Summary Log

```gams
model emp / master, submodel1, submodel2 /;

$onecho > "$emp.info"
  bilevel x1 x2 x3 x4
  min h1 * defh1 e1
  min h2 * defh2 e2
$offecho

solve emp us emp min z;
```

```plaintext
JAMS - Solver for Extended Mathematical Programs (EMP)

---------------
--- EMP Summary (errors=0)---
  Adjusted Constraint = 0
  Flipped Constraints = 0
  Dual Variable Maps = 0
  Dual Equation Maps = 0
  VI Functions = 0
  Equilibrium Agent = 0
  Bilevel Followers = 2
  Disjunctions = 0

The modelCritical.gms does not will be solved by GAMS.
```
EMP Library

- Distributed with GAMS
Summary

EMP

- automates symbolic reformulations to avoid error-prone and time-consuming manual algebra (re)writing
- offers solutions where solutions couldn’t be offered before
  • Embedded Complementarity Systems
  • Disjunctive Programs
  • Bilevel Programs
  • Extended Nonlinear Programs
  • Variational Inequalities
- facilitates to compare concurrent strategies
- free

- But: non-exhaustive, yet!
BETA 23.6
GAMS 23.6 Beta

Released November, 6th

www.gams.com/beta

• New Solver Libraries
  – COIN-OR (BONMIN 1.4, CSDP 6.1.1, …)
  – CPLEX 12.2.0.1
  – GUROBI 4.0
  – KNITRO 7.0
  – MOSEK 6.0.91
  – SCIP 2.0
  – XPRESS 21.01

• More solvers support in-core communication
  – Conopt, Knitro, Lgo, Mossek, Xpress
GAMS 23.6 Beta cont’d

Released November 6th

www.gams.com/beta

• Chk4Upd

• Python APIs to component libraries

• GAMSIDE updates

• XLSDump

• New library models (datalib, emplib, modlib, testlib)

• Internal Reorganization (non-linear instructions, hessian evaluators)
Contacting GAMS

**USA**

GAMS Development Corp.
1217 Potomac Street, NW
Washington, DC 20007

USA

Phone: +1 202 342 0180
Fax: +1 202 342 0181

[http://www.gams.com](http://www.gams.com)

sales@gams.com
support@gams.com

**Europe**

GAMS Software GmbH
Eupener Str. 135-137
50933 Cologne

Germany

Phone: +49 221 949 9170
Fax: +49 221 949 9171

[http://www.gams.de](http://www.gams.de)

info@gams.de
support@gams-software.com