GAMS
Model Development – Using CHP as an example

Michael Bussieck
mbussieck@gams.com
Jan-Hendrik Jagla
jhjagla@gams.com
Lutz Westermann
lwestermann@gams.com

GAMS Software GmbH
www.gams.de
GAMS Development Corporation
www.gams.com
## Agenda

<table>
<thead>
<tr>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAMS – Basic Syntax</td>
</tr>
<tr>
<td>Excursus: GDX</td>
</tr>
<tr>
<td>Building a Model: CHP Generation Plant</td>
</tr>
</tbody>
</table>
Agenda

GAMS – Basic Syntax

Excursus: GDX

Building a Model: CHP Generation Plant
GAMS Syntax: Declaration

- **Sets**

  ```gams
  Sets
  i       canning plants / seattle, san-diego /
  h       hours          / 1*24 /
  work(h) hours of work / 9*12, 14*17 /;
  ```

- **Parameters**

  ```gams
  Parameters
  a(i)   capacity of plant i in cases
          / seattle  350
          san-diego  600 /
  
  Table d(i,j) distance in thousands of miles
    new-york   chicago   topeka
  seattle  2.5   1.7   1.8
  san-diego 2.5   1.8   1.4
  ```

- **Scalars**

  ```gams
  Scalar f  freight in dollars per case per thousand miles /90/ ;
  ```
GAMS Syntax: Data Assignment using Sets

- **General**

  Parameter $c(i,j)$ transport cost in thousands of dollars per case;
  
  $c('seattle','chicago') = f * d('seattle','chicago') / 1000$;
  
  $c(i,j) = f * d(i,j) / 1000$;

- **Sum**

  Parameter daypay(i) Payment for a complete workday in $;
  
  daypay(i) = sum(h$work(h), pay(i,h));
  
  daypay(i) = sum(work(h), pay(i,h));
  
  daypay(i) = sum(work, pay(i,work));

- **Product**

  scalar prodcap Product of all capacities;
  
  prodcap = prod(i,a(i));

- **Minimum/Maximum**

  Scalar maxdem Maximum of all demands;
  
  maxdem = smax(j,b(j));

  Scalar mindist Minimum of all distances;
  
  mindist = smin((i,j),d(i,j));
GAMS Syntax: Defined Elements of a Set

• **Ord() and Card()**

  ```plaintext
  Set lasth(h) Last hour of the day;
  lasth(h) = (ord(h) = card(h));
  lasth(h)$ (ord(h) = card(h)) = yes;
  ```

• **Sameas(,)**

  ```plaintext
  Scalar demXny Demand in all markets except for New-York;
  demXny = sum(j$(not sameas(j,'new-york')), b(j));
  ```
GAMS Syntax: Variables

- Free ($-\infty$ to $\infty$)

```
Variables z Total transportation costs in thousands of dollars;
```

- Positive (0 to $\infty$)

```
Positive Variable X(i,j) Shipment quantities in cases;
```

- Negative ($-\infty$ to 0)

```
Negative Variable Y(h) Resource consumption;
```

- Integer (0, 1, 2, ...)

```
Integer Variable OUT(h) Output;
```

- Binary (0 or 1)

```
Binary Variable PRODUCE(i) Decision whether to produce or not;
```
GAMS Syntax: Variables

- Semi continuous (0 or above certain value)

  ```gams
  SemiCont Variable SHIP(i,j) Ship at least 100 tons;
  SHIP.lo(i,j) = 100;
  ```

- Semi integer (0 or integer above certain value)

  ```gams
  SemiInt Variable OUTP(i) Produce at least 12 units;
  OUTP.lo(i) = 12;
  ```

- Special Ordered Sets Type 1 (Only one member in a set of variables can have nonzero value)

  ```gams
  SOS1 Variable PRODUCE(i) Produce at one location only;
  ```

- Special Ordered Sets Type 2 (Only two adjacent members in a set of variables can have nonzero value)

  ```gams
  SOS2 Variable WORKSCED(h) Schedule work so that 2 hours in series are assigned;
  ```
GAMS Syntax: Equations

• Definition

```
Equations
   cost          define objective function
   supply(i)     observe supply limit at plant i
   demand(j)     satisfy demand at market j;
```

• Declaration

```
cost ..      Z  =e=  sum((i,j), c(i,j)*X(i,j));
supply(i) .. sum(j, X(i,j)) =l= a(i);
```
GAMS Syntax: Model Definition

- Model
  
  ```gams
  Model transport /all/;
  ```

- Solver selection
  
  ```gams
  option lp=coincbc;
  ```

- GAMS options
  
  ```gams
  Option reslim = 60;
  Option iterlim = 100;
  ```

- Solver options
  
  ```gams
  $onecho > cplex.opt
  lpmethod = 4
  $offecho
  ```

- Solve
  
  ```gams
  Solve transport using lp minimizing Z;
  ```
GAMS Syntax: Procedural Elements

• For

    scalar scen;
    for(scen=1 to 10 by 0.5,
        f = 10*scen;
        c(i,j) = f * d(i,j) / 1000;
        Solve transport using lp minimizing Z;
        Display Z.1;);

• While

    Scalar scen /1/;
    while(scen<=10,
        f = 10*scen;
        c(i,j) = f * d(i,j) / 1000;
        Solve transport using lp minimizing Z;
        scen = scen + 0.5;
    );

    loop(h,
        if(work(h),
            pay(i,h) = 0.6*pay(i,h);
        else
            pay(i,h) = 1.5*pay(i,h);
        );
    );

    pay(i,h) = 0.6*pay(i,h)$(work(h)) + 1.5*pay(i,h)$(not work(h));
## Mathematical Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>erf(x)</td>
<td>Integral of the standard normal distribution from $-\infty$ to $x$</td>
</tr>
<tr>
<td>exp(x)</td>
<td>Exponential, $e^x$</td>
</tr>
<tr>
<td>log(x)</td>
<td>Natural logarithm, $\log_e x$</td>
</tr>
<tr>
<td>log10(x)</td>
<td>Common logarithm, $\log_{10} x$</td>
</tr>
<tr>
<td>normal(x,y)</td>
<td>Random number normally distributed with mean $x$ and standard deviation $y$</td>
</tr>
<tr>
<td>uniform(x,y)</td>
<td>Random number with uniform distribution between $x$ and $y$</td>
</tr>
<tr>
<td>abs(x)</td>
<td>Absolute Value of $x$, i.e. $</td>
</tr>
<tr>
<td>ceil(x)</td>
<td>Ceiling of $x$. Smallest integer $\geq x$</td>
</tr>
<tr>
<td>floor(x)</td>
<td>Floor of $x$. Largest integer $\leq x$</td>
</tr>
<tr>
<td>mapval(x)</td>
<td>Mapping function. Assigns unique numbers to special values.</td>
</tr>
<tr>
<td>max(x,y,...)</td>
<td>Largest value among all arguments</td>
</tr>
<tr>
<td>min(x,y,...)</td>
<td>Smallest value among all arguments</td>
</tr>
<tr>
<td>mod(x,y)</td>
<td>Remainder. $x - y \cdot \text{trunc}(x/y)$</td>
</tr>
<tr>
<td>power(x,y)</td>
<td>Integer power. $x^y$, where $y$ must be an integer</td>
</tr>
<tr>
<td>round(x)</td>
<td>Round $x$ to the nearest integer</td>
</tr>
<tr>
<td>round(x,y)</td>
<td>Rounds $x$ to $y$ decimal places right (+) or left (-) to the decimal point</td>
</tr>
<tr>
<td>sign(x)</td>
<td>Returns 1 if $x &gt; 0$, -1 if $x &lt; 0$, and 0 if $x = 0$.</td>
</tr>
<tr>
<td>sqr(x)</td>
<td>Square of $x$. $x^2$</td>
</tr>
<tr>
<td>sqrt(x)</td>
<td>Square root of $x$. $\sqrt{x}$</td>
</tr>
<tr>
<td>trunc(x)</td>
<td>$\text{sign}(x) \cdot \text{floor}(\text{abs}(x))$</td>
</tr>
<tr>
<td>arctan(x)</td>
<td>$\tan^{-1} x$. Result in radians</td>
</tr>
<tr>
<td>cos(x)</td>
<td>$\cos x$; $x$ in radians</td>
</tr>
<tr>
<td>sin(x)</td>
<td>$\sin x$; $x$ in radians</td>
</tr>
</tbody>
</table>
Compile Time vs. Execution Time

• Compile time arguments…
  – start with $  
  – are executed when compiling a GAMS file  
  – are e.g. $if, $set, $goto, $exit, $call …

• Execution time arguments…
  – are executed during the execution of the compiled GAMS file  
  – are e.g. if, execute, solve, loop, …

NOTE: When reading a model from top to bottom, we can see an execution time command before a compile time command, but the latter will be executed first.
Agenda

- GAMS – Basic Syntax
- Excursus: GDX
- Building a Model: CHP Generation Plant
Gams Data eXchange

Binary Data Exchange

- Fast exchange of data
- Syntactical check on data before model starts
- Data Exchange at any stage (Compile and Run-time)
- Platform Independent
- Direct Excel connectivity
- General API
- Scenario Management Support

GDX Tools

- GDX Viewer
- GDXRank
- GDXMerge
- GDXDiff
- GDXAPI
- IDE
Using GDXXRW to read from Excel

Parameter d(i,j) distance in thousands of miles;

$call GDXXRW dist.xls par=d rng=dist!A1 rdim=1 cdim=1
$if errorlevel 1 $abort "Problem with file dist.xls!"
$gdxin dist
$load d
Using GDXXRW to write to Excel

```
execute_unload 'ship' x;
execute 'GDXXRW ship.gdx var=x rng=ship!A1 rdim=1 cdim=1';
```
GAMS – Basic Syntax

Excursus: GDX

Building a Model: CHP Generation Plant
Combined Heat and Power (CHP) Plant

- Produces heat and electricity in combination
- Certain demand of heat and electricity has to be satisfied
- Electricity can be traded at energy exchange
- Excess heat cannot be released, the demanded amount has to be generated exactly
- Cogeneration is subsidized by government
Process Model
Process Model

- External Inputs/Output
- Intermediate Commodities

<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>
</tr>
</thead>
<tbody>
<tr>
<td>1</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></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>Gas</td>
<td>-39.6</td>
<td>-85.67</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Waste Heat</td>
<td>53.2</td>
<td>-42.56</td>
<td>-10.64</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Steam</td>
<td>10.72</td>
<td>-16.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Exhaust</td>
<td></td>
<td>16.7</td>
<td>-16.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Heat</td>
<td>25</td>
<td>8.6184</td>
<td>39.84</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Electricity</td>
<td>29.55</td>
<td>10.39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Processes with minimum utilization level

<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>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HTB</td>
<td>HTB</td>
<td>GT</td>
<td>GT</td>
<td></td>
<td>HRB</td>
<td>HE</td>
<td>ST</td>
<td>HC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Gas</td>
<td>0</td>
<td>-39.6</td>
<td>-54.69</td>
<td>-85.67</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>MWh</td>
</tr>
<tr>
<td>4</td>
<td>WasteHeat</td>
<td>36.43</td>
<td>53.2</td>
<td>-42.56</td>
<td>-10.64</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MWh</td>
</tr>
<tr>
<td>5</td>
<td>Steam</td>
<td></td>
<td></td>
<td>10.72</td>
<td>-16.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>kg/s</td>
</tr>
<tr>
<td>6</td>
<td>Exhaust</td>
<td></td>
<td></td>
<td>16.7</td>
<td>-16.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>kg/s</td>
</tr>
<tr>
<td>7</td>
<td>Heat</td>
<td>0</td>
<td>25</td>
<td>13.77</td>
<td>29.55</td>
<td>8.6184</td>
<td>39.84</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MWh</td>
</tr>
<tr>
<td>8</td>
<td>Electricity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10.39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MWh</td>
</tr>
</tbody>
</table>
Demand Electricity/Heat
**Demand Electricity/Heat**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>HeatDemand</td>
<td>ElectricityDemand</td>
<td>EXPrice</td>
</tr>
<tr>
<td>2</td>
<td>h1</td>
<td>13.93</td>
<td>23.48</td>
<td>44.78</td>
</tr>
<tr>
<td>3</td>
<td>h2</td>
<td>12.98</td>
<td>22.64</td>
<td>36.14</td>
</tr>
<tr>
<td>4</td>
<td>h3</td>
<td>12.35</td>
<td>21.91</td>
<td>31.02</td>
</tr>
<tr>
<td>5</td>
<td>h4</td>
<td>11.60</td>
<td>21.35</td>
<td>27.96</td>
</tr>
<tr>
<td>6</td>
<td>h5</td>
<td>12.58</td>
<td>20.81</td>
<td>26.61</td>
</tr>
<tr>
<td>7</td>
<td>h6</td>
<td>12.08</td>
<td>20.34</td>
<td>25.55</td>
</tr>
<tr>
<td>8</td>
<td>h7</td>
<td>10.41</td>
<td>19.37</td>
<td>16.00</td>
</tr>
<tr>
<td>9</td>
<td>h8</td>
<td>9.13</td>
<td>19.66</td>
<td>19.17</td>
</tr>
<tr>
<td>10</td>
<td>h9</td>
<td>8.88</td>
<td>19.38</td>
<td>26.00</td>
</tr>
<tr>
<td>11</td>
<td>h10</td>
<td>9.19</td>
<td>19.02</td>
<td>37.62</td>
</tr>
<tr>
<td>12</td>
<td>h11</td>
<td>8.96</td>
<td>18.66</td>
<td>46.83</td>
</tr>
<tr>
<td>13</td>
<td>h12</td>
<td>9.52</td>
<td>18.38</td>
<td>57.79</td>
</tr>
<tr>
<td>14</td>
<td>h13</td>
<td>10.43</td>
<td>18.14</td>
<td>53.98</td>
</tr>
<tr>
<td>15</td>
<td>h14</td>
<td>12.38</td>
<td>17.92</td>
<td>46.04</td>
</tr>
<tr>
<td>16</td>
<td>h15</td>
<td>13.97</td>
<td>17.74</td>
<td>41.19</td>
</tr>
<tr>
<td>17</td>
<td>h16</td>
<td>14.83</td>
<td>17.68</td>
<td>38.51</td>
</tr>
<tr>
<td>18</td>
<td>h17</td>
<td>14.64</td>
<td>17.69</td>
<td>46.46</td>
</tr>
<tr>
<td>19</td>
<td>h18</td>
<td>14.19</td>
<td>17.65</td>
<td>64.10</td>
</tr>
<tr>
<td>20</td>
<td>h19</td>
<td>13.61</td>
<td>17.56</td>
<td>71.51</td>
</tr>
<tr>
<td>21</td>
<td>h20</td>
<td>12.92</td>
<td>17.57</td>
<td>67.81</td>
</tr>
<tr>
<td>22</td>
<td>h21</td>
<td>12.46</td>
<td>17.35</td>
<td>55.24</td>
</tr>
<tr>
<td>23</td>
<td>h22</td>
<td>12.02</td>
<td>16.96</td>
<td>50.46</td>
</tr>
<tr>
<td>24</td>
<td>h23</td>
<td>11.27</td>
<td>16.79</td>
<td>57.59</td>
</tr>
<tr>
<td>25</td>
<td>h24</td>
<td>11.50</td>
<td>16.90</td>
<td>45.75</td>
</tr>
</tbody>
</table>
Modeling Task

- Find cost minimal solution
  - Satisfy demand
  - Buy or make electricity
  - Subsidize cogeneration
  - Technical feasible schedule of plants
  - Investment decisions (new/upgraded power plants)
  - Economical aspects (e.g. shared ownership of plants)
  - ...

Minimize [Fuel costs]
   + [Costs/returns from electricity trading]
   − [Bonus for cogeneration]

s.t.       [Matter input] = [Matter output]

            [Generated electricity] + [Purchased electricity]
            = [Demand for electricity]

            [Generated heat] = [Demand for heat]
Exercise 1: Add Steam Generator (SG)

- At maximum utilization:
  - Output: Steam 11 kg/s
  - Input: Coal 35.24 MWh
  Electricity 0.5 MWh

- At minimum utilization:
  - Output: Steam 5.5 kg/s
  - Input: Coal 17.62 MWh
  Electricity 0.25 MWh

- Coal costs: 12.23 $/MWh

- Reduces Output of HTB: 25 MWh → 18 MWh
Exercise 2: Add Heat Bypass (HB)

- Consumes up to 53.2 MWh wasteheat
- Cools it down at costs of 4 $ per MWh
- No relevant output
Exercise 3: Add Heat Storage Tank (HST)

- Maximum capacity of 50 MW heat
- At most 15 MW per hour input
- At most 12 MW per hour output
- “Pump” heat into tank costs 0.05 $ per MW
- 2% of stored heat gets lost per h
Heat Storage Tank

Before:

\[ dem_{Heat}(h) = GEN_{Heat}(h) \]

After:

\[ dem_{Heat}(h) = GEN_{Heat}(h) + HOUT(h) - HIN(h) \]

\[ HLVL(h) = HLVL(h-1) \cdot 0.98 - HOUT(h) + HIN(h) \]

\[ HLVL(h) \leq 50 \]

\[ HIN(h) \leq 15 \]

\[ HOUT(h) \leq 12 \]
Exercise 4: Limiting Number of GT Starts

- Startup costs:
  - GT: 500 $
  - SG: 1000$
- GT may be turned on not more than 8 times during modeled time frame

\[ \text{ONOFF}(h, p) = 1 \land \text{ONOFF}(h-1, p) = 0 \Rightarrow \text{STARTUP}(h, p) = 1 \]
\[ \Rightarrow \text{STARTUP}(h, p) \geq \text{ONOFF}(h, p) - \text{ONOFF}(h-1, p) \]

\[ \sum_{h} \text{STARTUP}(h, 'GT') \leq 8 \]
Exercise 5: Add “cool down” time for GT

- GT has to stay off for at least 8 hours when shut down

\[
\text{ONOFF}(h, p) = 0 \land \text{ONOFF}(h - 1, p) = 1 \implies \text{SHUTDOWN}(h, p) = 1 \\
\implies \text{SHUTDOWN}(h,'GT') \geq \text{ONOFF}(h - 1,'GT') - \text{ONOFF}(h,'GT')
\]

\[
\text{SHUTDOWN}(h,'GT') = 1 \\
\implies \text{STARTUP}(h_2,'GT') = 0 \mid h \leq h_2 < h + 8 \\
\implies \sum_{h_2 \mid h_2 \geq h \land h_2 < h + 8} \text{STARTUP}(h_2,'GT') \leq 1 - \text{SHUTDOWN}(h,'GT')
\]
Demo: Calling GAMS from MS Excel

<table>
<thead>
<tr>
<th></th>
<th>SG</th>
<th>SG</th>
<th>HTB</th>
<th>GT</th>
<th>GT</th>
<th>HRB</th>
<th>HE</th>
<th>HB</th>
<th>ST</th>
<th>HC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gas</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Coal</td>
<td></td>
<td>17.62</td>
<td>-54.69</td>
<td>85.67</td>
<td></td>
<td>MWh</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WasteHeat</td>
<td>36.43</td>
<td>53.2</td>
<td>-42.56</td>
<td>10.64</td>
<td>-53.2</td>
<td>MWh</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Steam</td>
<td>5.5</td>
<td>11</td>
<td>10.72</td>
<td>-16.7</td>
<td></td>
<td>kg/s</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exhaust</td>
<td>18</td>
<td>8.6184</td>
<td>39.84</td>
<td></td>
<td></td>
<td>MWh</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

GAMS Directory: `C:\program files\gams22.8\`
Working Directory: `C:\tmp`
Solver: CPLEX

<table>
<thead>
<tr>
<th>COINPOPT</th>
<th>CONOPT</th>
<th>CPLEX</th>
<th>KNITRO</th>
<th>LINDO</th>
<th>GLOBAL</th>
<th>MINOS</th>
<th>MOSEK</th>
<th>XPRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
GAMS on the Web

Download  www.gams.de  www.gams.com

Help and Support
Support Wiki  http://support.gams-software.com
Interfaces Wiki  http://interfaces.gams-software.com
User Group  http://www.gams.com/maillist/gams_l.htm
Google Group  http://groups.google.de/group/gamsworld

Search all GAMS Websites  http://www.gams.com/search.htm
Contacting GAMS

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

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
sales@gams.com
support@gams.com