AIMMS

An All-Round Development Environment

Jan Bisschop

Paragon Decision Technology B.V.
Haarlem, The Netherlands
Presentations and Demos

Outline:

• Paragon and AIMMS
• AIMMS Outer Approximation Approach
• Quantities and Units
• Database Interaction
• Multi-Language Support
• Excel Interface
• Multi-Agent Technology
Paragon Decision Technology B.V.

- Dutch company, founded in 1989
- Activities
  - AIMMS development
  - Modeling support and consulting
- AIMMS major releases
  - Version 3.x : 1999 - 2006
  - Version 4.x : 2007 -
- Web site: www.aimms.com
AIMMS Acronym

Advanced
Integrated
Multi-Dimensional
Modeling
Software
AIMMS Overview

- Point & Click
- Modeling Language / Model Explorer
- Cplex, Conopt, Xpress, etc.

- GUI Construction
- Model Building
- Solvers

- External DLLs
  - Fortran / C

- Databases
  - ODBC / OLE DB

- AIMMS API / COM Object
AIMMS Productivity Tools

- **Model Explorer**
  - tree-based model construction
- **Identifier Selector**
  - customizable model overviews
- **Page Manager**
  - navigational interface structure
- **Template Manager**
  - page and template construction
- **Menu Builder**
  - customizable menus
- **Data Manager**
  - case management
- **Data Management Setup**
  - advanced case management
AIMMS-Based DSS Applications

- Clearing Energy Markets (LP, ALSTOM ESCA)
- Cat Cracking Model (NLP, Shell Oil)
- Advanced Blending Module (LP+MIP+NLP, Shell)
- Data Reconciliation (NLP, Shell Chemicals)
- Crude Oil Scheduling (Discrete Event, Shell)
- Refinery Scheduling (NLP, Amoco)
- Forestry Management System (LP, Ontario)
- Structured Asset Management (LP+NLP, ABN-AMRO)
- Scheduling Beer Production (Heuristic, Heineken)
- Simulation Chemical Plant (Simulation, EC Germany)
- Cargo Revenue Management (Lufthansa Cargo)
- Liquidation Planning System (Brown-Forman)
- Management Chain Game (CentER Appl. Research)
AIMMS-based DSS Applications
Interface Example (Cat Cracker Model)
AIMMS-based DSS Applications
Interface Example (Crude Oil Scheduling)
Standard Features

• (Finite) Index Sets
  – Compound
  – Indexed
  – Integer

• Parameters
  – Numerical-valued
  – Element-valued
  – String-valued
  – Unit-valued

• Variables and Constraints

• (Multiple) Mathematical Programs
  – Linear and Mixed-Integer Linear
  – Nonlinear and Mixed-Integer Nonlinear
  – Mixed Complementarity
Standard Features Continued

• Procedures and Functions
  – Intrinsic
  – Internal
  – External

• Flow Control
  – If-then-else
  – For / While / Repeat
  – Switch
  – Halt

• Optimization
  – Solve

• Data Communication
  – Read / Write
Key Features

• Sparse Execution and Data Storage
• Procedures and Definitions Combined
• Global Index Domain Propagation
• Database Communication
• Statistical / Financial / GUI Functions
• XML Support
• Solver Callbacks
Unique Features

- **End-User GUI Builder**
- **Units of measurement**
  - automatic unit checking in model
  - unit conventions for switching in end-user interface
- **Modeling of time**
  - horizons for period-based modeling
  - calendars for calendar-based interfacing
  - flexible support for aggregation and disaggregation
- **White Box Outer Approximation**
- **Multi-Agent Technology**
Transportation Demo

- Inspect Model Structure
- Extend the Model
- Introduce Template Page
- Copy Existing Page
- Adjust Graphical Network Object
- Introduce Page Navigation Buttons
- Specify Page Resize Behavior
AIMMS Outer Approximation for MINLP

Min z = c^T y + f(x)

s.t.

A y + h(x) = 0

B y + g(x) ≤ 0

C y + D x ≤ d

x ∈ X = \{ x ∈ R^n | x_L ≤ x ≤ x_U \}

y ∈ Y = \{ 0, 1 \}^m
Outer Approximation Solution Procedure

Solve the initial (relaxed) NLP model

Repeat

Generate and solve master MIP model
Solve the NLP sub-model
Check termination criteria
Prepare for next iteration

End Repeat
Solver Communication

Solver
- Interface
  - Exported Functions
  - Callbacks
- Generated Model(s)
- Algorithm(s)

AIMMS
- MP.cb := P;
- solve MP;
- procedure P
- display ..;
Open AOA Solver Communication

AOA Solver

- Interface
  - Exported Functions
  - Callbacks

- Initial NLP
- Master MIP
- NLP Subproblem
- CPLEX (MIP)
- CONOPT (NLP)

AIMMS

```
MP.OAcb := P;
solve MP;
```

```
procedure P

AOA algorithm
```
Outer Approximation Solution Procedure in AIMMS

Reset all state parameters
Solve the initial NLP problem

Repeat

Generate and solve master MIP problem
Solve the NLP subproblem
Check termination criteria
Prepare for next iteration

Endrepeat
### Outer Approximation Solution Procedure in AIMMS

The control procedure implements the actual outer approximation algorithm, and interacts with the Aimms Outer Approximation solver interface to solve and/or modify the NLP and master MIP subproblems.

You may modify the implementation of the control procedure as you see fit, or perhaps create a completely different control procedure.
Outer Approximation Interface
Exported Functions

• **Selected NLP Functions**
  - NLPSolve
  - NLPIsFeasible
  - NLPSolutionIsInteger

• **Selected Master MIP Functions**
  - MasterMIPSolve
  - MasterMIPSolveToNextIntegerNode
  - MasterMIPIsFeasible
  - MasterMIPAddIntegerSolutionEliminationCut
  - MasterMIPAddLinearizations
  - MasterMIPSetCallback

• **Selected Solution Manipulation Functions**
  - SolutionSave
  - SolutionRetrieve
  - SolutionReplace
  - SolutionDelete
Outer Approximation
Possible Uses of Openness

• User-specified switch between NLP solvers
• User-specified control (through MIP callback) over intermediate integer solutions for later evaluation
• User-specified points for adding linearizations
• User-specified procedures to adjust the penalties associated with linearizations in the Master MIP objective function
AIMMS Outer Approximation for Extended MINLP

\[ \text{Min } z = f(x,y) \]
\[ \text{s.t.} \]
\[ h(x,y) = 0 \]
\[ g(x,y) \leq 0 \]
\[ Cy + Dx \leq d \]
\[ x \in X = \{ x \in \mathbb{R}^n \mid x^L \leq x \leq x^U \} \]
\[ y \in Y = I^m \]
AIMMS Outer Approximation

Reactor Production Scheduling

20 – Product Scenario

- AIMMS NLP Branch & Bound
  - Solver did not terminate
  - First integer solution 32.53 after 25 seconds
  - Best integer solution 32.13 after 250 seconds
  - Optimal integer solution not found

- AIMMS Outer Approximation
  - 20 integer solutions in 500 seconds
  - Best integer solution 32.01 after 85 seconds
  - Optimal integer solution not found
DEMO
AIMMS Measurement Concepts

• Quantities (Expresses what is measured)
  - Basic (e.g. Length, Time)
  - Derived (e.g. Velocity)

• Units (Expresses a reference measurement)
  - Base Units (e.g. [m], [s], [m / s])
  - Compound Units (e.g. [Hz] = [1 / s])
  - Derived Units (e.g. [mile] -> [m] : # -> # *1609)

• Unit-valued Parameters
  - NutrientUnit(n) := data { Energy : [KJ],
    Protein : [mg], Iron : [%RDA] }
AIMMS Measurement Concepts

- **Unit-valued Parameters**
  - NutrientUnit(n) := data { Energy : [KJ],
    Protein : [mg], Iron : [%RDA] }
  - FoodUnit(f) := data { Cola : [liter],
    Carrots : [pounds], Burgers : [-] }

- **Numerical Parameter**
  identifier : NutrientValue
  index domain : (f,n)
  unit : NutrientUnit(n) / FoodUnit(f)
Unit Consistency

• Parameter
  identifier : A
  unit : [m / s]

• Parameter
  identifier : B
  unit : [km / hour]

• Parameter
  identifier : C
  unit : [mile / hour]
  definition : A + B + 3 [ km / s ]
Unit Support

- Wizard that uses database with
  - SI Quantities
  - Common Units
  - Decimal Scaling
AIMMS Link with Databases

- AIMMS connects to ODBC / OLE DB-compliant databases (Access, SQL Server, Oracle, etc.)
- AIMMS supports the use of
  - Database tables,
  - Database queries
  - Database procedures
  - Direct execution of SQL statements
Link Specification

• A Data Source

• A Table Name

• A Mapping of
  – primary key columns to AIMMS index domain
  – data columns to AIMMS identifiers
## Database Table

<table>
<thead>
<tr>
<th>Location</th>
<th>XCoordina</th>
<th>YCoordina</th>
<th>UnitStockCost</th>
<th>InitialStockLevel</th>
<th>MinimumStockLevel</th>
<th>MaximumStockLevel</th>
<th>NumberOfInhabitants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amersfoort</td>
<td>5.377</td>
<td>52.158</td>
<td>0.07</td>
<td>2208</td>
<td>460</td>
<td>2760</td>
<td>123367</td>
</tr>
<tr>
<td>Amsterdam</td>
<td>4.88</td>
<td>52.376</td>
<td>0.07</td>
<td>14016</td>
<td>2920</td>
<td>17520</td>
<td>727053</td>
</tr>
<tr>
<td>Apeldoorn</td>
<td>5.953</td>
<td>52.213</td>
<td>0.07</td>
<td>2736</td>
<td>570</td>
<td>3420</td>
<td>152860</td>
</tr>
<tr>
<td>Arnhem</td>
<td>5.917</td>
<td>51.982</td>
<td>0.07</td>
<td>2352</td>
<td>490</td>
<td>2940</td>
<td>137222</td>
</tr>
<tr>
<td>Assen</td>
<td>6.561</td>
<td>53.011</td>
<td>0.07</td>
<td>1056</td>
<td>220</td>
<td>1320</td>
<td>57376</td>
</tr>
<tr>
<td>Breda</td>
<td>4.778</td>
<td>51.588</td>
<td>0.07</td>
<td>2928</td>
<td>610</td>
<td>3660</td>
<td>159042</td>
</tr>
<tr>
<td>Den Bosch</td>
<td>5.308</td>
<td>51.701</td>
<td>0.07</td>
<td>2208</td>
<td>460</td>
<td>2760</td>
<td>128009</td>
</tr>
<tr>
<td>Den Haag</td>
<td>4.303</td>
<td>52.079</td>
<td>0.07</td>
<td>8064</td>
<td>1680</td>
<td>10080</td>
<td>440743</td>
</tr>
<tr>
<td>Den Helder</td>
<td>4.754</td>
<td>52.958</td>
<td>0.07</td>
<td>1056</td>
<td>220</td>
<td>1320</td>
<td>59590</td>
</tr>
<tr>
<td>Deventer</td>
<td>6.158</td>
<td>52.263</td>
<td>0.07</td>
<td>1488</td>
<td>310</td>
<td>1860</td>
<td>82621</td>
</tr>
<tr>
<td>Dordrecht</td>
<td>4.679</td>
<td>51.794</td>
<td>0.07</td>
<td>2160</td>
<td>450</td>
<td>2700</td>
<td>119462</td>
</tr>
<tr>
<td>Eindhoven</td>
<td>5.461</td>
<td>51.432</td>
<td>0.105</td>
<td>8330</td>
<td>8330</td>
<td>33320</td>
<td>199877</td>
</tr>
<tr>
<td>Emmen</td>
<td>6.885</td>
<td>52.788</td>
<td>0.07</td>
<td>1920</td>
<td>400</td>
<td>2400</td>
<td>105497</td>
</tr>
<tr>
<td>Enschede</td>
<td>6.89</td>
<td>52.22</td>
<td>0.07</td>
<td>2784</td>
<td>580</td>
<td>3480</td>
<td>148814</td>
</tr>
<tr>
<td>Groningen</td>
<td>6.574</td>
<td>53.226</td>
<td>0.07</td>
<td>3024</td>
<td>630</td>
<td>3780</td>
<td>171193</td>
</tr>
<tr>
<td>Haarlem</td>
<td>4.618</td>
<td>52.382</td>
<td>0.105</td>
<td>4310</td>
<td>4310</td>
<td>17240</td>
<td>148262</td>
</tr>
<tr>
<td>Leeuwarden</td>
<td>5.782</td>
<td>53.212</td>
<td>0.07</td>
<td>1632</td>
<td>340</td>
<td>2040</td>
<td>88762</td>
</tr>
<tr>
<td>Maastricht</td>
<td>5.696</td>
<td>50.857</td>
<td>0.07</td>
<td>2256</td>
<td>470</td>
<td>2820</td>
<td>121479</td>
</tr>
<tr>
<td>Nijmegen</td>
<td>5.845</td>
<td>51.84</td>
<td>0.07</td>
<td>2496</td>
<td>520</td>
<td>3120</td>
<td>151864</td>
</tr>
<tr>
<td>Rotterdam</td>
<td>4.482</td>
<td>51.929</td>
<td>0.07</td>
<td>10272</td>
<td>2140</td>
<td>12840</td>
<td>592665</td>
</tr>
<tr>
<td>Tilburg</td>
<td>5.071</td>
<td>51.568</td>
<td>0.07</td>
<td>3312</td>
<td>690</td>
<td>4140</td>
<td>190559</td>
</tr>
<tr>
<td>Utrecht</td>
<td>5.118</td>
<td>52.107</td>
<td>0.07</td>
<td>4080</td>
<td>850</td>
<td>5100</td>
<td>232718</td>
</tr>
<tr>
<td>Venlo</td>
<td>6.158</td>
<td>51.374</td>
<td>0.07</td>
<td>1200</td>
<td>250</td>
<td>1500</td>
<td>64580</td>
</tr>
<tr>
<td>Vlissingen</td>
<td>3.571</td>
<td>51.458</td>
<td>0.07</td>
<td>768</td>
<td>160</td>
<td>960</td>
<td>44530</td>
</tr>
<tr>
<td>Zwolle</td>
<td>6.09</td>
<td>52.522</td>
<td>0.105</td>
<td>2790</td>
<td>2790</td>
<td>11160</td>
<td>104431</td>
</tr>
</tbody>
</table>
Mapping Columns to Identifiers

```
<table>
<thead>
<tr>
<th>Identifier</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>LocationTable</td>
<td>&quot;Location&quot; --&gt; 1,</td>
</tr>
<tr>
<td></td>
<td>&quot;XCoordinate&quot; --&gt; XCoordinate( 1 ),</td>
</tr>
<tr>
<td></td>
<td>&quot;YCoordinate&quot; --&gt; YCoordinate( 1 ),</td>
</tr>
<tr>
<td></td>
<td>&quot;InitialStockLevel&quot; --&gt; StockAtStartOfCalendar( 1 ),</td>
</tr>
<tr>
<td></td>
<td>&quot;MinimumStockLevel&quot; --&gt; MinimumStock( 1 ),</td>
</tr>
<tr>
<td></td>
<td>&quot;MaximumStockLevel&quot; --&gt; MaximumStock( 1 ),</td>
</tr>
<tr>
<td></td>
<td>&quot;UnitStockCost&quot; --&gt; UnitStockCost( 1 )</td>
</tr>
</tbody>
</table>
```

"Softdrink Planning"

"Locations"
Database Procedure

**NumberOfProductionLinesQuery**

<table>
<thead>
<tr>
<th>Database procedure</th>
<th>NumberOfProductionLinesQuery</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arguments</td>
<td></td>
</tr>
<tr>
<td>Data source</td>
<td>&quot;Softdrink Planning&quot;</td>
</tr>
<tr>
<td>Sql query</td>
<td>&quot;SELECT Factory, COUNT(ProductionLine) AS LineCount &quot; + &quot;FROM ProductionLines GROUP BY Factory&quot;</td>
</tr>
<tr>
<td>Owner</td>
<td>UseResultSet</td>
</tr>
<tr>
<td>Property</td>
<td></td>
</tr>
<tr>
<td>Mapping</td>
<td>&quot;Factory&quot; --&gt; f, &quot;LineCount&quot; --&gt; NumberOfProductionLines(f)</td>
</tr>
<tr>
<td>Convention</td>
<td></td>
</tr>
<tr>
<td>Comment</td>
<td></td>
</tr>
</tbody>
</table>
AIMMS Multi-language Support

- AIMMS Unicode: support for double-byte (i.e. Asian, Russian) character sets
- Easy selection of language in end-user interface
- Support for automatic extension of existing models with language database
AIMMS and Spreadsheets Combined

AIMMS
- model
- algorithms
- GUI

Excel
- data
- GUI

AIMMS in control : Excel Interfacing Functions
Excel in control : AIMMS Excel Add-In

Paragon Decision Technology B.V.
Excel Add-In Features

• Data Transfer between AIMMS and Excel
  — Retrieve Set/Array/Table Data
  — Assign Set/Array/Table Data

• Empty AIMMS Identifiers

• Update AIMMS Identifiers

• Run AIMMS Procedures

• Run Execution Sequence

• Run Excel Macro
The AIMMS
Multi-Agent Technology
What is a Multi-Agent System?

• An agent is a software program that independently performs its tasks on behalf of its user.

• A multi-agent system is a community of agents that communicate together to accomplish individual and/or common goals.
How do Agents Communicate?

Agent 1

Request

Response

Agent 2

Request

Response

Agent 3

MQ 1

MQ 2

MQ 3
What are Agent Characteristics?

Agents are ...

- **reactive:** they react to messages
- **proactive:** they initiate messages
- **intelligent:** they can make decisions
- **autonomous:** they control their own actions and internal states
Computing Paradigm

- **Parallel computing**: agents can reside on different computers.
- **Dynamic computing**: agents can enter and leave the community dynamically.
- **Non-deterministic computing**: the order of execution is not known a priori.
Example: A Purchasing Agent

A purchasing agent independently:

- checks inventories at regular moments
- determines order quantities
- requests price information from suppliers
- places orders
- updates the administration
- etc.
Example: Auctions (B2B)

Bidding agents:
- supply bids each round
- decide on how to modify the bids
- know when to stop

Market clearing agent:
- receives the bids each round
- informs the bidding agents
- determines final allocation
Example: Control Systems

Different specialized agents:

- to observe and register plant measurements
- to analyze these measurements
- to decide when and what to communicate with other agents
- to exercise control when needed
Example: Algorithms

Worker agents to solve:

• sub-models in branch-and-bound
• experiments in Monte Carlo optimization
• scenario’s in stochastic programming
Example: Pooled Resource Allocation

- Independent contractors (agents)
  - generate improved resource use schedules

- Pooled resource coordinator (agent)
  - selects one schedule per contractor
  - minimizes total resource cost
  - updates resource price
### Available

- 80 men at 100 $/man
- 50 men at 150 $/man

### Men in Schedule

<table>
<thead>
<tr>
<th></th>
<th>Company 1</th>
<th>Company 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>50</td>
<td>70</td>
</tr>
<tr>
<td>Feb</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>Mar</td>
<td>30</td>
<td>20</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$/man</th>
<th>$/man</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>117</td>
<td>117</td>
</tr>
<tr>
<td>Feb</td>
<td>110</td>
<td>110</td>
</tr>
<tr>
<td>Mar</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$/man</th>
<th>$/man</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>112</td>
<td>108</td>
</tr>
<tr>
<td>Feb</td>
<td>106</td>
<td>105</td>
</tr>
<tr>
<td>Mar</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$/man</th>
<th>$/man</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>55</td>
<td>55</td>
</tr>
<tr>
<td>Feb</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>Mar</td>
<td>40</td>
<td>40</td>
</tr>
</tbody>
</table>

### Summary

- Total Available: 30000
- Total Available in Schedule: 28750
- Total Available Excess: 1250

(14400, 15600)

(13800, 14950)

(13800, 14700)
AIMMS Agent Projects

- Each AIMMS project contains one or more individual agents.
- Projects have a physical location in a local-area network.
- Each project has its own associated message queue.
- There is a hierarchy between projects to start and stop other projects and their associated queues.
Individual AIMMS Agents

• Each individual AIMMS agent has its own role in an agent-based application.
• Several agents can have the same role.
• For each role there is an associated set of types of messages to be handled.
• For each type of message received, there is a separate AIMMS procedure to handle that particular type of message.
Agent Community Setup

Paragon Decision Technology B.V.
Specific Agent Setup