The Role and Impact of Algebraic Modeling Languages in and on Industrial Optimization

Michael R. Bussieck: MBussieck@gams.com
Franz Nelissen: FNelissen@gams.com
Lutz Westermann: LWestermann@gams.com
Company Background

Roots: World Bank, 1976

GAMS Development Corporation (Washington)

Tool Provider: General Algebraic Modeling System

Went commercial in 1987

GAMS Software GmbH (Cologne, Braunschweig) 1996
Agenda

- Algebraic Modeling Languages – A Success Story
- GAMS – Design Principles
- Some Examples
The Vision

RESULT:
- Limited drain of resources
- Same representation of models for humans and machines
- Model representation is also model documentation
Algebraic Modeling Languages (AML)

1. High-level computer programming languages
   - Formulation of mathematical optimization problems
   - Notation similar to algebraic notation

2. Do not solve problems directly, but offer links to state-of-the-art algorithms (“solver-links”)

Source: http://en.wikipedia.org/wiki/Algebraic_modeling_language
2012 INFORMS Impact Prize

Originators of Algebraic Modeling Languages

36 Years later
Impact of Algebraic Modeling Languages

1. Tremendous increased productivity, quality, reliability and maintainability
2. Simplified model development, changes, and transfer
3. Made a scarce resource (good modelers) more productive

Important vehicle to make mathematical optimization available to a broader audience
Agenda

Algebraic Modeling Languages – A Success Story

GAMS – Design Principles

Some Examples
Design Principles

1. Simple modeling language with a balanced mix of declarative and procedural elements

2. Open architecture and interfaces to other systems

3. Independent layers
Simple Declarative Language

1. Language similar to mathematical notation
2. Few basic language elements: sets, parameters, variables, equations, models → Easy to learn
3. Lot’s of code optimization under the hood
Mix of Declarative and **Procedural Elements**

- Procedural elements like loops, for, if, macros and functions
  - Allow to build complex problem algorithms within GAMS
  - Interaction with other systems:
    - Job control
    - Data exchange

**Combine models inside the system**
Independence of Model and Operating System

Platforms supported by GAMS:

- Microsoft Windows
- Linux
- MacOS
- Solaris
- AIX

Models can be moved between platforms with ease
Independence of **Model and Solver**

**One** environment for a wide range of model types and solvers

- All major commercial LP/MIP solver
- Open Source Solver (COIN)
- Also solver for NLP, MINLP, global, and stochastic optimization

> **Switching between solvers with one line of code**
Independence of **Model and Data**

- **Declarative Modeling:** \( x(j), j \in \{1,...\} \)
  - ASCII: Initial model development
  - GDX: Binary Data layer ("contract") between GAMS and applications
    - Platform independent
    - Direct GDX interfaces and general API

-GAMS-
Independence of Model and User Interface

1. Open architecture and interfaces to other systems → No preference for a particular user interface

2. Application Programming Interfaces
   - Low Level
   - Object Oriented: .Net, Java, Python, ...

3. Smart Links to popular environments
   - Excel, MATLAB, R, ...

Model
Platform  Solver  Data  Interface
### Broad Range of Application Areas

<table>
<thead>
<tr>
<th>Agricultural Economics</th>
<th>Applied General Equilibrium</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chemical Engineering</td>
<td>Economic Development</td>
</tr>
<tr>
<td>Econometrics</td>
<td>Energy</td>
</tr>
<tr>
<td>Environmental Economics</td>
<td>Engineering</td>
</tr>
<tr>
<td>Finance</td>
<td>Forestry</td>
</tr>
<tr>
<td>International Trade</td>
<td>Logistics</td>
</tr>
<tr>
<td>Macro Economics</td>
<td>Military</td>
</tr>
<tr>
<td>Management Science/OR</td>
<td>Mathematics</td>
</tr>
<tr>
<td>Micro Economics</td>
<td>Physics</td>
</tr>
</tbody>
</table>

25+ Years GAMS Development
Broad User Community and Network

GAMS used in more than 120 countries

25+ Years GAMS Development
Broad User Community and Network

More than 10,000 licenses

50% academic users, 50% commercial users

6,000+ monthly downloads of the free system

25+ Years GAMS Development
Agenda

Algebraic Modeling Languages – A Success Story

GAMS – Design Principles

Some Examples
“... - asking yourself why you would do optimization in GAMS, software that looks as modern and friendly as Richard Nixon.”

Three (Short) Examples

1. General Dynamics – retooling the solution engines for the generation and management of efficient frontiers of multi-objective design optimization problems

2. xyz Energy Company – solving many (500+) difficult MIP models in parallel on the cloud (security issues)

3. Class Scheduling at USMA West Point – student centric scheduling of classes
Advanced Collaborative System Optimization Modeler

Explore high-dimensional Pareto surface for configuration of (expensive) military vehicles

Collaboration between:
- General Dynamics Land Systems
- Industrial & Systems Engineering, Wayne State University (WSU)
- GAMS Development Corp
- Amsterdam Optimization Group
The Whole-system Design Problem

21 SUBSYSTEMS

- AFES
- Armor
- Chassis Structure
- Crew Station
- Defensive Armament
- Dismountable
- ECS
- Fuel
- Hit Avoidance
- Lighting
- Mission Equipment (Shooter)
- Mission Station
- Mission Structure
- NBC
- Platform Electronics
- Power Distribution & Mgmt
- Propulsion
- Signature Management
- Suspension
- Turret Structure
- Water Management

<table>
<thead>
<tr>
<th>Options per subsystem</th>
<th>Theoretically Possible Subsystem Combinations</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2,092,152</td>
</tr>
<tr>
<td>3</td>
<td>10,460,353,203</td>
</tr>
</tbody>
</table>

Approved for Public Release, Distribution Unlimited, GDLS approved, log 2008-09, dated 03/19/08 4
Considers All to Make the Whole

Full Spectrum

Numerous Subsystems with a multitude of options for each

Force Protection Approach?
Two, Three, Four Crew Members?
Auto loader?
  • Configuration/Approach?
Hull Material?
  • Aluminum?
  • Steel?
  • Titanium?
Which Core Data Network?

Power?
  • What Type of Engine?
  • What Type Transmission?
What Type of Suspension System?
  • Torsion bars?
  • HSUs?
  • Passive?
  • Track?
  • Fully Active?

Which Servo Motor Controller Architecture?

Approved for Public Release, Distribution Unlimited, GDLS approved, log 2008-09, dated 03/19/08
Utopia Point

alias(thisobj,obj);

loop(thisobj, w(obj) = 0;
    w(thisobj) = 1;

solve mod1 using mip max ztotal;
f_best(thisobj) = z.l(thisobj);

solve mod1 using mip min ztotal;
f_worst(thisobj) = z.l(thisobj);

f_range(thisobj) = f_best(thisobj) - f_worst(thisobj);
);

Compromise Point

equation
distance 'distance to Utopia point';

variable
    sqdist 'minimum distance (normalized)';

distance.. sqdist =e= sum(obj, sqr((f_best(obj)-z(obj))/f_range(obj)));

model mindist /all/;
solve mindist minimizing sqdist using miqcp;

How to calculate Efficient Frontier?
Many Algorithms

- Complete Enumeration
- Evolutionary Algorithms
- WSU Algorithms: ACE, UPSA
- Integer Cut

AC SOM Algorithm

- GAMS/CPLEX solution pool to collect 1,000,000+ solutions of a small MIQP
- Compact representation of solutions
- Uses an algorithm to filter out dominated solution
  - implemented in traditional programming language with efficient data structures
Retooling of an Application

Old System
- VB.Net
- MPL (single point)
- CPLEX
- SQL throughout

New System
- GAMS (complete algorithm; streamlined design)
- GAMS .NET API for recursive ACE algorithm
- CPLEX
- Bulk SQL just once

Performance Improvements
- On the same hardware:
  - From hours and even days for large problems
  - To minutes, up to an hour
xyz Energy Company

Massive Parallel for Large Scale MIP

- Sensitivity Analysis subject to few parameter changes
- Failures with Stochastic Optimization and Robust Optimization
- Scenario Analysis works!
- Task: Solve 1000 scenarios (MIPs, 1-2 hours) every week overnight
- Post processing (distribution of KPI) of results outside modeling system
Solving Independent Models

- Small Ratio of MP solution time/AML overhead
  - GUSS/Scenario Solver
- Large ratio i.e. only MP time is relevant (pre/post processing not critical)
  - Grid Facility, Gurobi/Cplex Server
- If entire model run including pre processing/MP solve/post processing is costly
  - Parallel/asynchronous execution of entire model in the cloud
  - Security issues in the cloud
    - Obfuscation
Massively Parallel MIP

• MIP/B&C Algorithm ideal to parallelize
  • Master/Worker Paradigm (process nodes in parallel)
  • Software: FATCOP/Condor, BCP/PVM, PICO/MPI
• Distributed MIP (sub-MIPs in parallel)
  • Software: ParaSCIP, Cplex (12.x), Gurobi (6.0)
• A-priori subdivision into n independent problems
  • Seymour problem solved that way
• Open Pit Mining (openpit in GAMS Model library)
  • Partitioning integer variables to subdivide model into 4096 sub-problems
• Experiments (Ferris) at UW using Condor Pool
HTCondor (previously Condor)

Computing with HTCondor™

Our goal is to develop, implement, deploy, and evaluate mechanisms and policies that support High Throughput Computing (HTC) on large collections of distributively owned computing resources. Guided by both the technological and sociological challenges of such a computing environment, the Center for High Throughput Computing at UW-Madison has been building the open source HTCondor distributed computing software (pronounced “aitch-tee-condor”) and related technologies to enable scientists and engineers to increase their computing throughput.

Note: The HTCondor software was known as ‘Condor’ from 1988 until its name changed in 2012. If you are looking for Phoenix Software International’s software development and library management system for z/VSE or z/OS, click here.

Our annual user conference, HTCondor Week is scheduled for April 28-30, 2014. Save the date!
Results for 4096 MIPS on HTCondor

- Submission started Jan 11, 16:00
- All jobs submitted by Jan 11, 23:00
- All jobs returned by Jan 12, 12:40
  - 20 hours wall time, 5000 CPU hours
  - Peak number of CPU’s: 500
AWS Cloud Setup at xyz Company
Configure the Worker Instance

- Standard EC2 Linux image
- Prepare mounting the S3 bucket on instance start up
- Install GAMS
- Starting the Python xmlrpc server on start up
- Add the following lines to `/etc/rc.d/rc.local`

```
sudo cd /usr/share
sudo mkdir s3fs && cd s3fs
sudo wget http://s3fs.googlecode.com/files/s3fs-1.63.tar.gz
sudo tar -xvf s3fs-1.63.tar.gz
sudo yum update all
sudo yum install gcc libstdc++-devel gcc-c++ fuse fuse-devel curl-devel libxml2-devel openssl-devel mailcap
cd s3fs-1.63
sudo ./configure --prefix=/usr
sudo make
sudo make install
```

- mount the S3 bucket
- start the daemon with the RCP server
RPC Server – Run at Machine Start Up

- Start of server

```python
server = SimpleXMLRPCServer.SimpleXMLRPCServer(("0.0.0.0", 8080))
server.register_function(spawn)
server.register_function(get_result)
server.register_function(is_done)
server.serve_forever()
```

- Asynchronous run of GAMS Job

```python
def spawn(scen):
    t = threading.Thread(target=run_gams, args=(scen + 1,))
    t.start()
```

- GAMS OO API to control GAMS Job

```python
def run_gams(scen):
    ws = GamsWorkspace(system_directory=sysdir)
    cp_from = ws.add_checkpoint('/mys3bucket/indus89a')
    cp_to = ws.add_checkpoint('/mys3bucket/' + str(scen) + '_result')
    job = ws.add_job_from_string(get_model_text(), checkpoint=cp_from)
    opt = ws.add_options()
    opt.defined["scenumber"] = str(scen)
    opt.license = '/mys3bucket/gamslice.txt'
    job.run(opt, checkpoint=cp_to)
```
Submitter - Managing EC2 instances

- **Start instances**
  ```python
  # get EC2 connection and EC2 image
  conn = EC2Connection(access_key_id, secret_access_key)
  image = conn.get_all_images([image_id])

  # start n instances
  instances = image[0].run(min_count=n, max_count=n,
  security_groups=["All Incoming"],
  instance_type='t1.micro').instances
  ```

- **Check status**
  ```python
  for s in scen:
      instances[s].update()
      if instances[s].state == "running":
  ```

- **Terminate instances**
  ```python
  # shut down all machines after global time out has expired
  for s in scen:
      instances[s].update()
      if instances[s].state not in ["shutting-down", "terminated"]:
          instances[s].terminate()
  ```
Submitter - Communicating via RPC

- Establish connection
  ```python
  server = xmlrpclib.Server("http://" + instances[s].ip_address + ":8080",
  transport=TimeoutTransport())
  servers[s] = server
  ```

- Call functions provided by the RPC server
  - Start job on server
    ```python
    server.spawn(s)
    print "Scenario " + str(s+1) + ": Job was spawned successfully."
    ```

- Check status and collect results
  ```python
  if servers[s].is_done():
    results[s] = servers[s].get_result()
    print "Scenario " + str(s+1) + ": Getting results and terminating instance"
  ```
Protecting IP and Sensitive Data

- Obfuscate or hide sensitive information
  - Extrinsic function libraries
  - External Equations
  - Secure work files
  - Encrypted source files
  - Obfuscated files
    - change all the names and other documentation related to a specific model run
Obfuscated Model - Example

- GAMS Model Library: indus89 + scenario data
  ```gams
  Set demandscen;
  Parameter demandfactor(demandscen,cq);
  $gdxin demandfactor
  $load demandscen demandfactor
  loop(demandscen$(ord(demandscen)=%scenumber%),
      demand(z,cq) = demand(z,cq) * demandfactor(demandscen,cq));
  ```
- Create obfuscated save file (compile only)
  - gams indus89 action=compile so=indus89A
- Send to cloud and execute
  - gams runscn –scennumber=n r=indus89A s=n
- Copy save files 1,...,n back and create named results
  - gams result –scennumber=n r=n rn=indus89
- Merge results (e.g. gdxmerge)
  - gdxmerge *_result.gdx
Class Scheduling at USMA

1999:
- Class Scheduling at United States Military Academy, West Point, NY, USA
- Soon after Term End Exam Scheduling (TEE) at USMA

“... each student’s daily activities are a carefully regimented balance of academic, military, and physical requirements.”

2014:
- Applications are alive and kicking
- New UI & computing platform
USMA is Different

• Technically
  • Day1/day2 schedule
  • Special rules (e.g. < 30% athletes in class)
• Scheduling around the cadets needs
  • No conflicting activities
  • Individual schedule of activities is compliant to vast catalogue of business rules
• Proper planning
ATTENDANCE PERIODS

1-DAY CLASS PERIODS

0735 0830 0840 0935 0945 1040 1050 1145 1245 1340 1350 1445 1455 1550

A  B  C  D  NOON MEAL  CMDT’S HOUR  E  F

LAB  LAB  LAB

2-DAY CLASS PERIODS

0735 0830 0840 0935 0945 1040 1050 1145 1245 1340 1350 1445 1455 1550

G  H  I  J  NOON MEAL  DEAN’S HOUR  K  L

LAB  LAB  LAB

R/S/T/U LAB HOURS
USMA is Different

- Technically
  - Day1/day2 schedule
  - Special rules (e.g. < 30% athletes in class)
- Scheduling around the cadets needs
  - No conflicting activities
  - Individual schedule of activities is compliant to vast catalogue of business rules
- Proper planning
Academic Program

- 8 Term Academic Program (8TAP)
- For example, Mathematics Major: Statistics/Optimization

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PH202</td>
<td>ENG SEQ</td>
<td>ENG SEQ</td>
<td>ENG SEQ</td>
<td>ENG SEQ</td>
<td>ENG SEQ</td>
<td>ENG SEQ</td>
<td>ENG SEQ</td>
</tr>
<tr>
<td>2</td>
<td>MA206</td>
<td>ENG SEQ</td>
<td>MA363</td>
<td>MA386</td>
<td>MA476*</td>
<td>MA476*</td>
<td>MA476*</td>
<td>MA476*</td>
</tr>
<tr>
<td>3</td>
<td>LX2XX</td>
<td>MA376</td>
<td>MA371</td>
<td>MA381*</td>
<td>MA481*</td>
<td>MA481*</td>
<td>MA481*</td>
<td>MA481*</td>
</tr>
<tr>
<td>4</td>
<td>SS20X</td>
<td>MA383</td>
<td>MA387</td>
<td>MA492D *</td>
<td>MA491</td>
<td>MA491</td>
<td>MA491</td>
<td>MA491</td>
</tr>
<tr>
<td>5</td>
<td>EV203</td>
<td>SS307</td>
<td>EN302</td>
<td>HI301</td>
<td>HI302</td>
<td>HI302</td>
<td>HI302</td>
<td>HI302</td>
</tr>
<tr>
<td>6</td>
<td>MA 391</td>
<td>PL300</td>
<td>LW403</td>
<td>LW403</td>
<td>LW403</td>
<td>LW403</td>
<td>LW403</td>
<td>LW403</td>
</tr>
</tbody>
</table>
Enterprise Academy Management System

- Management of:
  - Redbook
  - Registration
  - Schedule
  - Facility
  - Attendance
  - Grades
Academic Scheduling

Course Scheduling
- For a given set of course offerings find good schedules for all cadets.
- Scheduling preps
- Term balancing (department resources)
- Initial course offerings
- Scheduling
- Manual post scheduling

Term End Exam (TEE)
- Find good schedules for exam courses and cadets
### MA104 Course Offerings

<table>
<thead>
<tr>
<th>COURSE NBR</th>
<th>SECT</th>
<th>SECT SIZE</th>
<th>CPBLTY</th>
<th>ENRLD</th>
<th>SEATS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA104</td>
<td>52</td>
<td>18</td>
<td>936</td>
<td>933</td>
<td>3</td>
</tr>
</tbody>
</table>

### MA104 Instruction Periods

<table>
<thead>
<tr>
<th>COURSE NBR</th>
<th>HOUR</th>
<th>SECT</th>
<th>SECT SIZE</th>
<th>CPBLTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA104</td>
<td>AG</td>
<td>13</td>
<td>18</td>
<td>234</td>
</tr>
<tr>
<td>MA104</td>
<td>BH</td>
<td>13</td>
<td>18</td>
<td>234</td>
</tr>
<tr>
<td>MA104</td>
<td>CI</td>
<td>13</td>
<td>18</td>
<td>234</td>
</tr>
<tr>
<td>MA104</td>
<td>DJ</td>
<td>13</td>
<td>18</td>
<td>234</td>
</tr>
</tbody>
</table>

### MA205 Course Offerings

<table>
<thead>
<tr>
<th>COURSE NBR</th>
<th>SECT</th>
<th>SECT SIZE</th>
<th>CPBLTY</th>
<th>ENRLD</th>
<th>SEATS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA205</td>
<td>5</td>
<td>18</td>
<td>90</td>
<td>70</td>
<td>20</td>
</tr>
</tbody>
</table>

### MA205 Instruction Periods

<table>
<thead>
<tr>
<th>COURSE NBR</th>
<th>HOUR</th>
<th>SECT</th>
<th>SECT SIZE</th>
<th>CPBLTY</th>
</tr>
</thead>
<tbody>
<tr>
<td>MA205</td>
<td>AG</td>
<td>1</td>
<td>18</td>
<td>18</td>
</tr>
</tbody>
</table>
Scheduling Cadet Courses to Hours

Given course hours & capacity

- MA481,AB,36
- MA481,CD,18
- MA481,EF,18
- PE300,C,180
- PE300,J,60
- MA371,F,18

Given cadet’s course registration

- 043671571,MA481
- 043671571,PE300

Objective: Find a good assignment of cadet’s course requests to course hours

- 043671571,MA481,CD
- 043671571,PE300,J
Scheduling System Objectives

- Produced in a timely fashion
- Complies with *most* constraints/rules...
  - Individual:
    - Cadet can’t be at two places at the same time
    - Course hours are balanced over Day-1, Day-2
    - Free hour
    - Cadet gets scheduled for all courses he/she has registered for
  - Capacity:
    - Enrollment of course hour cannot exceed capability
    - Course enrollment of freshmen which are also athletes cannot exceed 60% of the total enrollment
Legacy Scheduling System

- Matching algorithm based on cadet’s course registration only
- Reporting programs check schedule against all constraints and business rules
- Last years (1998): 80% of cadets rescheduled manually
- Time to generate schedule: 4 weeks (3 Schedulers)
An Optimization Model

\[
\begin{align*}
\text{min} & \sum_{ro} (p1_{ro} \pi 1_{ro} + p2_{ro} \pi 2_{ro}) + \sum_{c} (p3_c \pi 3_c + p4_c \pi 4_c) \\
\sum_{o} x_{c,ro} &= 1 \quad \text{(for all 8TAP entries)} \\
\sum_{r} x_{c,ro} &\leq 1 + \pi 3_c \quad \text{(for all cadets c for all time slots o)} \\
-\sigma - \pi 4_c &\leq \sum_{ro \text{ on day-1}} x_{c,ro} - \sum_{ro \text{ on day-2}} x_{c,ro} \leq \sigma + \pi 4_c \quad \text{(for all cadets c)} \\
x_{c,ro} &= 0 \quad \text{(for all c,ro where c has activity at o)} \\
\sum_{c} x_{c,ro} &\leq cap_{ro} + \pi 1_{ro} \quad \text{(for all course hours ro)} \\
\sum_{c \text{ freshmen\&athlete}} x_{c,ro} - 0.6 \sum_{c} x_{c,ro} &\leq \pi 2_{ro} \quad \text{(for all course hours ro)}
\end{align*}
\]

• 60,000 Variables, 500,000 Non-Zeros
• 24 hours CPLEX 6.6 and no integer solution
Problems with a MP Model

• There is no solution subject to all constraints/rules for real data

• What is the objective for the model?

• *Goal Programming*:
  • Relax constraints/rules by penalizing violations. Select penalties for violations: Calibration of the model.
A 2-Stage Approach

- Prescheduling
  - Filter cadets with no feasible schedule
  - Overcome infeasibility by relaxation/data changes
- Scheduling
  - All individual constraints/rules are hard constraints
  - Find assignment that does not exceed capacity (or penalize overloads)
Prescheduling

One cadet at a time
- Check feasibility
- If infeasible produce several infeasible schedules ranked by severity of infeasibility
  - Hour Conflict
  - Day – Day Balance
  - Last Hour Free
- Human intervenes
Scheduling in Batches

- Now individual constraints allow feasible schedules for all cadets (hard constraints)
- Tight Capacity of Course Hours may still result in an infeasible model: 
  ➔ Assign Overload Penalties
- Model with all cadets still unsolvable!
Scheduling

- Take a batch of cadets (10-30) and schedule them independent of the others
- Objective: Meet the capacity profile of the course hours (Penalize Over- and Underload)
- Example:
  - 3 course hours MA481 AB (20), CD, IJ (10)
  - Batch contains 8 cadets with request for MA481
## Results

- AY 2000/2 tested in parallel
- AY 2001/1 deployed

<table>
<thead>
<tr>
<th></th>
<th>Legacy System</th>
<th>New System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual Relaxations</td>
<td>203/304/116</td>
<td>58/25/4</td>
</tr>
<tr>
<td>free/group/unbalanced</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Capacity Overloads Courses/seats</td>
<td>12/54</td>
<td>9/21</td>
</tr>
<tr>
<td>Number of Schedulers</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Time to produce Schedule</td>
<td>4 Weeks</td>
<td>1 Day</td>
</tr>
</tbody>
</table>
Summary/Some Lessons Learned

- Know what you can add to a project (modeling, optimization, algorithms)
- For successful application know more than math/modeling or work with competent people that know their stuff
- Use as much off-the-shelf products AML, Excel, Matlab, R, ...
- Visualization of results
EMS-EDM Prophet
GENERAL ALGEBRAIC MODELING SYSTEM

EMS-EDM PROPHET® – Modules 2015

© 2014 Fraunhofer IOSB-AST
EMS-EDM PROPHET® – Anwender

© 2014 Fraunhofer IOSB-AST
EMS-EDM PROPHET® – Übersicht

- ZRM: Zeitreihenmanagement
- ZPS: Zeitreihenprognosesystem
- RPS: Ressourcenplanungssystem
- BKM: Bilanzkreismanagement
- NNM: Netznutzungsmanagement
- NMS: Netzmanagementsystem
- SDL: Scheduler

Mathematical Optimization
Optimierung
- Kraftwerkseinsatz
- Querverbund
- Virtuelle Kraftwerke
- Beschaffung
- Handel
- Regelenergie
- Speicher

Zahlreiche Modelltypen
- LP, MIP, QMIP etc.
- NLP, MINLP etc.

Solver
- CPLEX, GUROBI etc.
- CONOPT, LINDO etc.
Modellkomponenten für skalierbare Systeme (Beispiele)

Thermische Kraftwerksblöcke
- Max. Leistungsänderung
- Min. und max. Energievorgaben
- Minutenspannen aus Stillstand
- Pos. und neg. Sekundäreleistungsvorhandenheit im Zeitintervall
- Kauf/Verkauf
  - Spotmarkt/Börse Strom/Gas

Hydraulische Kraftwerksblöcke
- Max. Pumpleistung im Zeitintervall
- Min. und max. Beckenstandsgeführt
- Max. Betriebsstunden
- KWK-Förderung
- Gasbefeuierung (Sterlingmotor)
- Wärmespeicher
- Füllstandsbeschränkungen
- Verfügbarkeiten
  - Wärmebedarf
  - Endverbraucher

Verfügbare Energien
- Verfügbarkeiten
  - BHKW
  - Vorgaben für technische Revisionen
  - Berücksichtigung der aktuellen Erzeugung bei Planung

© 2014 Fraunhofer IOSB-AST
RPS – Modeling

Graphical modeling of optimization problems

Drag 'n Drop

Power plant
RPS - Modeling
Lade Optimierungs-DLL
Operation abgeschlossen 00:00
Lade Zeitbasis
Operation abgeschlossen: 00:00
Erzeuge Projektverzeichnis und entferne temporäre Dateien
Operation abgeschlossen: 00:00
Erzeuge GAMS-Datei ...
schreibe Komponentendeklarationen
schreibe GDX-Daten
schreibe Instanzwerte
schreibe Set-Datenwerte
schreibe Set-Datenwerte für die Startbedingungen
schreibe Tabellen
schreibe Tabelle "Startzustandstabelle [NW_BK_034]"
...
Zeitbedarf GDX-Datei-Erzeugung [hh:mm:ss]: 00:00:09
schreibe Modelldaten
schreibe Instanzgleichungen
schreibe Bilanzgleichungen
schreibe Untermodellgleichungen
schreibe Objektfunktion
Schreiben der Deklarationsdatenwerte
Schreiben der Deklarationswerte abgeschlossen
Erzeuge Optionsdatei (OptFile)
Zeitbedarf GAMS-Datei-Erzeugung [hh:mm:ss]: 00:02:34
...
Durchführung der Optimierung ...
[19.03.2014 01:00 - 23.03.2014 00:00]
Datei: "C:\GAMS\win64\24.3\gams.exe"
Parameter: "C:\EMS-EDM PROPHET\system\Rps-
EMS-EDM PROPHET–RPS

- Optimierung
  - Kraftwerkseinsatz
  - Querverbund
  - Virtuelle Kraftwerke
  - Beschaffung
  - Handel
  - Regelenergie
  - Speicher

- Zahlreiche Modelltypen
  - LP, MIP, QMIP etc.
  - NLP, MINLP etc.

- Solver
  - CPLEX, GUROBI etc.
  - CONOPT, LINDO etc.
Fields of Fuel

A Multiplayer, Web-based Simulation Game
FoF allow classes to explore the complex sustainability challenges associated with growing bioenergy crops.
Sustainable Bioenergy Crop Production

<table>
<thead>
<tr>
<th>SUSTAINABILITY CATEGORY</th>
<th>SCORE COMPONENT</th>
<th>SCORE SUBCOMPONENTS (DATA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOCIAL</td>
<td>ENERGY PRODUCTION</td>
<td>• CROP YIELD (MEGAGRAMS OF BIOMASS)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NET ENERGY (MegaJoules)</td>
</tr>
<tr>
<td>ECONOMIC</td>
<td>INCOME</td>
<td>• NET INCOME ($)</td>
</tr>
<tr>
<td>ENVIRONMENTAL</td>
<td>ENVIRONMENTAL QUALITY</td>
<td>• SOIL FERTILITY (SOIL CARBON)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WATER QUALITY (INDEX)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• BENEFICIAL BUGS (INDEX)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GREENHOUSE GAS EMSSIONS (CO2 EQ.)</td>
</tr>
</tbody>
</table>
## Sustainable Bioenergy Crop Production

<table>
<thead>
<tr>
<th>Crop Choices</th>
<th>Economics</th>
<th>Energy/Yield</th>
<th>Environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corn</td>
<td>High</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Alfalfa</td>
<td>Medium</td>
<td>Low (N/A)</td>
<td>High</td>
</tr>
<tr>
<td>Switchgrass</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
</tr>
</tbody>
</table>
Fields of Fuel – Setting up a Game

Game Controls:
- Start
- End Game
- Help
- Apply Changes

Game Settings:
- Market-Driven Prices Enabled: [ ]
- Management Options Enabled: [ ]
- Help Popups Disabled: [ ]

Adjust Crop Prices:
- Corn: $250
- Grass: $80
- Alfalfa: $125

Game Options:
- Robot Opponents: 3
- # of Fields: 4
- Variable Initial Soil Health: [ ]

[Begin] [Help]
Fields of **Fuel – GAMS robots**

- Single player have the option of choosing the number of bots to play against; the player can also set the objective of each bot opponent.
- The robots optimize a weighted average of overall score and field health over a five year period.
- A mixed-integer nonlinear programming problem (MINLP) identifies the optimal future path to follow.
Fields of Fuel
Fields of Fuel
Fields of Fuel

- Sustainability: 69% Avg Score
- Economics: 99% Avg Score
- Energy: 63% Avg Score
- Environment: 43% Avg Score

Field Changes Over Time
- Economy
- Energy
- Environment

Overall Water Emissions
- Farm emissions
- Average emissions
Fields of Fuel

fieldsoffuel.org

Wisconsin Institute for Discovery
Summary

AML – A Success Story

GAMS - Design Principles

- Simple, but powerful language for optimization problems
- Open interfaces
- Different layers

Examples

- Provide cutting edge technology
- Don’t lock developers and users into a certain environment
- Protect investments of users
- Make mathematical optimization available to a broader audience
Thank You

USA
GAMS Development Corp. 1217 Potomac Street, NW Washington, DC 20007
USA
Phone: +1 202 342 0180
Fax: +1 202 342 0181
sales@gams.com

Europe
GAMS Software GmbH
P.O. Box 40 59
50216 Frechen, Germany
Phone: +49 221 949 9170
Fax: +49 221 949 9171
info@gams.de