

GAMS Models in Agricultural Economics

Wolfgang Britz

Institute for Food And Resource Economics
University Bonn

- Simulation models in agricultural economics
- CAPRI as an example: what, why and how
- Why GAMS: features, costs, user community
- Limitations of GAMS
- CAPRI GUI: wrapping GAMS
- Summary



- Long tradition:
 - 1950ties: LP based **farm level optimization**, today mostly MIP, fully dynamic, stochastic
 - 1970ties: application to „**typical**“ or „**regional**“ **farm** for policy analysis
 - Branch dried out for a while due to typical problems of LPs (jumpy behaviour, over-specialization) and hunger for data not available from statistics



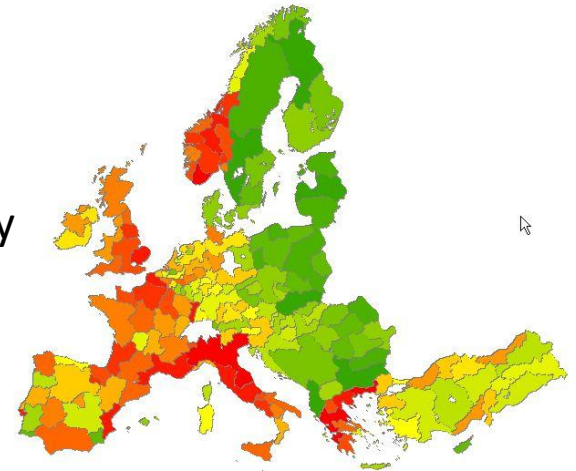
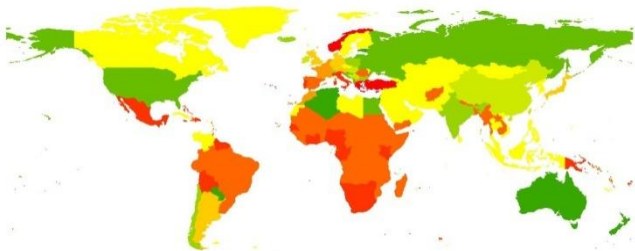
- 1980ties, **price endogenous market models**:
 - **Multi-Commodity** models:
 - Global simultaneous equilibrium for many agricultural products and many countries/country blocks
 - Prices, supply, demand and trade endogenous
 - Non-linear CNS or NCP
 - **Computable General Equilibrium** Models:
 - Simultaneous equilibrium for all markets in an economy (input, outputs, primary factors)
 - Non-linear CNS or MCP



- Today:
 - **Revival of programming models**
 - introduction of (econometrically estimated) non-linearities eases calibration and avoids over-specialization
 - increasing focus on agri-environmental interactions (bio-economic modelling)
 - All model types **grow in resolution** (products, space, agents) and **complexity** (policy instrument covered, functional forms, stochastics ..)
 - **Combination** of different type **of models** into tools for impact assessment

- **Common Agricultural Policy Regionalized Impact:**
 - Tool for **impact assessment** of policies related to the agricultural sector
 - Clients: mainly **EU commission**, but also national government and industry
 - Combines economic models, matching data bases, IT infrastructure and institutional setting
 - Financed mainly by EU research framework programs and tenders

- Two modules, links by sequential calibration
 - About 2000 **farm type models** with
(each ~300 vars x ~200 equations, NLPs, independently solved as grid)

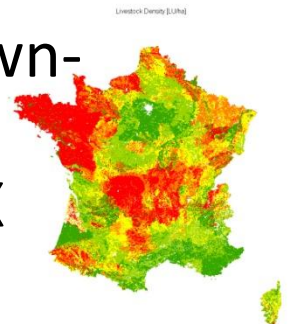


- **Global, spatial trade model** for agricultural products
(CNS, in parts highly non linear, 40.000x40.000)

- One run solves each module ~20 times, ~20-30 minutes, generates ~15 Mio non zeros, in parts post-model calculated economic and environmental indicators

- Simulates **impacts of policy instruments** such as tariffs, subsidies, agri-environmental programs
- On **various indicators** such as
 - Crop areas, yields, herd sizes, production, feed use
 - Human consumption, industrial processing, trade flows
 - Farm income, EU budget costs, consumer welfare
 - GHG emissions, nutrient balances
- Developed end of the nineties, since then continuously expanded, improved and applied
- **Open source** concept, **network** based

- Complete work flow in GAMS
 - **Data base generation** and **model parameterization**:
 - Mapping/aggregation/dis-aggregation of raw official data
 - Outlier detection and correction
 - Bayesian estimators to consolidate raw statistical data
 - Estimators for behavioural parameters
 - Generation of „**market outlook**“ (currently 2020)
 - Model **calibration** to outlook results
 - **Simulation** runs
 - **Post model processing**, including statistical down-scaling to about 200.000 1x1 km clusters
 - Results are passed between worksteps via GDX





- Developed originally **from economists for economists**:
 - **Notation** comes close to **scientific papers**
 - Relatively **easy to learn**
 - **Compact notation** for element wise operations, avoids explicit loop and if statements
 - **Transparent interface to high performance solvers** for different problem types: LP, NLP, MIP, CNS, MCP etc.

- Basic **IDE**, sufficient for medium sized projects
- **Proprietary binary data format (GDX)**:
 - Fast, saves disk space, tailored for large sparse matrices, link e.g. to EXCEL
 - API support for high-level programming languages, provides bridge to DBMS or tailored GUIs
- Continuous development e.g. support for new solvers

- Relatively **restricted language features**
 - no functions/sub-routines
 - limited number of object classes (set, parameter, equation, variable, model, file ..)
 - all numbers are double precisions (not distinctions between int, long, float, double etc.)
 - no explicit string handling
 - every symbol has global scope



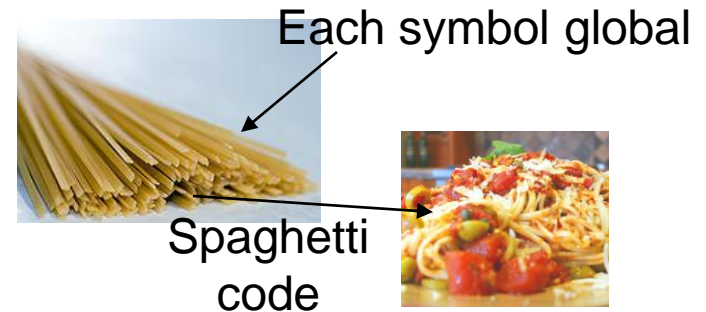
- **Consequences** of restricted language features:
 - **no formal education in IT required** – fits typical profile of (agr) economists
 - debugging is typical easy
 - **low learning costs**
 - **dis-advantages** for **experienced programmers** and large-scale projects

- **Trial version** (often sufficient for didactic purposes) can be downloaded for free
- Relatively **modest license fees** for degree granting institutions
- Good **manuals**, active **e-mail list**
- Long list of **example models**
- **Portable** between platforms, can be used without GUI/IDE

- Features explain wide spread use in community as a kind of „**lingua franca**“:
 - Taught in **courses at master level** based on „toy models“, e.g. already in Germany by various ag-econ departments
 - Allows researchers to switch relatively easy between tools and institutions
 - Eases tool linkage

- Many other examples of policy relevant ag-econ models realized in GAMS, e.g.
 - Uni Hohenheim: ESIM
 - Univ. Wageningen, NL: farm type models
 - Texas A&M: FASOM
 - Penn state: PEATSIM
 - Purdue: GTAP in GAMS
 - ...

- **Modular design** and **exchange of code** between projects **not easy**:
 - **Global scope** of all symbols, only **rudimentary** support for **functions/subroutines**
 - ↔ namespace conflicts
 - ↔ no encapsulation
 - ↔ documentation is tedious
 - IDE not ideal for large-scale projects compared e.g. with Eclipse



- So far **no parallel execution in the base module** and in some widely solvers such as CONOPT:
 - Multiple cores not used
 - Pre- and post model processing increasingly CPU bottleneck
- Parallel model solves supported, but I/O intensive
 - Little gain for a suite of small model instances
 - „hot“ updates in solvers in early stages, applicability not yet clear

- GAMS alone not appropriate to host complex tools, CAPRI e.g. should allow
 - users to easily **define, start** and **exploit own scenarios** (if possible without touching GAMS directly)
 - administrators to **steer supporting work steps** (such as data base and outlook generation, calibration), best without coding
 - efficiently analysis of **huge data sets**

- CAPRI responded by developing its **own GUI**
 - Realized in **Java**
 - **Steers GAMS** work steps by generating „code snippets“ and spawning GAMS
 - comprises powerful **exploitation tools** (reports as tables, graphs, maps) to analyze results
 - access to GAMS generated data via API bridge to GDX



Switch modules on/off

What to do (step & task)

Post-model reporting options

Start GAMS

The screenshot shows the CAPRI GUI interface with the following components:

- Work step selection:** A vertical list of radio buttons including 'Build database', 'Generate baseline', 'Run scenario' (selected), 'Collect meta information', 'Batch execution', 'Generate GAMS documentation', and 'Exploit.gdx files'.
- Task selection:** A vertical list of radio buttons including 'Define scenario', 'Run scenario' (selected), 'Downscale scenario results', and 'Exploit scenario results'.
- Task properties for : run scenario:**
 - Base year: 2004
 - Simulation year: 2020
 - Countries: BL (Belgium & Luxembourg), DK (Denmark), DE (Germany), EL (Greece), ES (Spain)
 - Regional break down: NUTS 2
 - Number of iterations: 50
 - Scenario definition file: MTR_RDG20.GMS
 - Market models:
 - Global market model
 - Young animal market model
 - Regional CGEs
 - Reporting:
 - Aggregates for activities and commodities
 - Environmental indicators
 - Life-cycle assessment for energy
 - Multi-functionality indicators
 - Iteration tracking
- Buttons:** 'compile GAMS', 'run GAMS', 'show results', and 'show meta data'.
- Status Bar:** CAPRI GUI Version 3.0, August 20...; Ini file : capri.ini; User name : Wolfgang Britz; User type : administrator

Work step is running ...
(GUI has spawned GAMS)

GAMS
output redirected to GUI

The screenshot shows the CAPRI GUI interface. The top window, titled 'CAPRI [t:\britz\capri\gams]', displays 'Task properties for: Run scenario'. The properties are as follows:

- Base year: 2004
- Simulation year: 2020
- Countries: BE (Belgium & Luxembourg), DK (Denmark), DE (Germany), EL (Greece)
- Regional break down: NUTS 2
- Number of iterations: 50
- Scenario definition file: MTR_NDG20NOPIL1.GMS
- Market models:
 - Global market model
 - Young animal market model
 - Regi
- Reporting:
 - Aggregates for activities and commodities
 - Environmental indicators
 - Life-
 - Multi-functionality indicators
 - Iteration tracking

Below the properties is a 'Stop GAMS' button. The main window displays the GAMS output for 'CAPMOD: Solve market, year 2020, step 4(50), (9.49 0/0), try # 0'. The output includes:

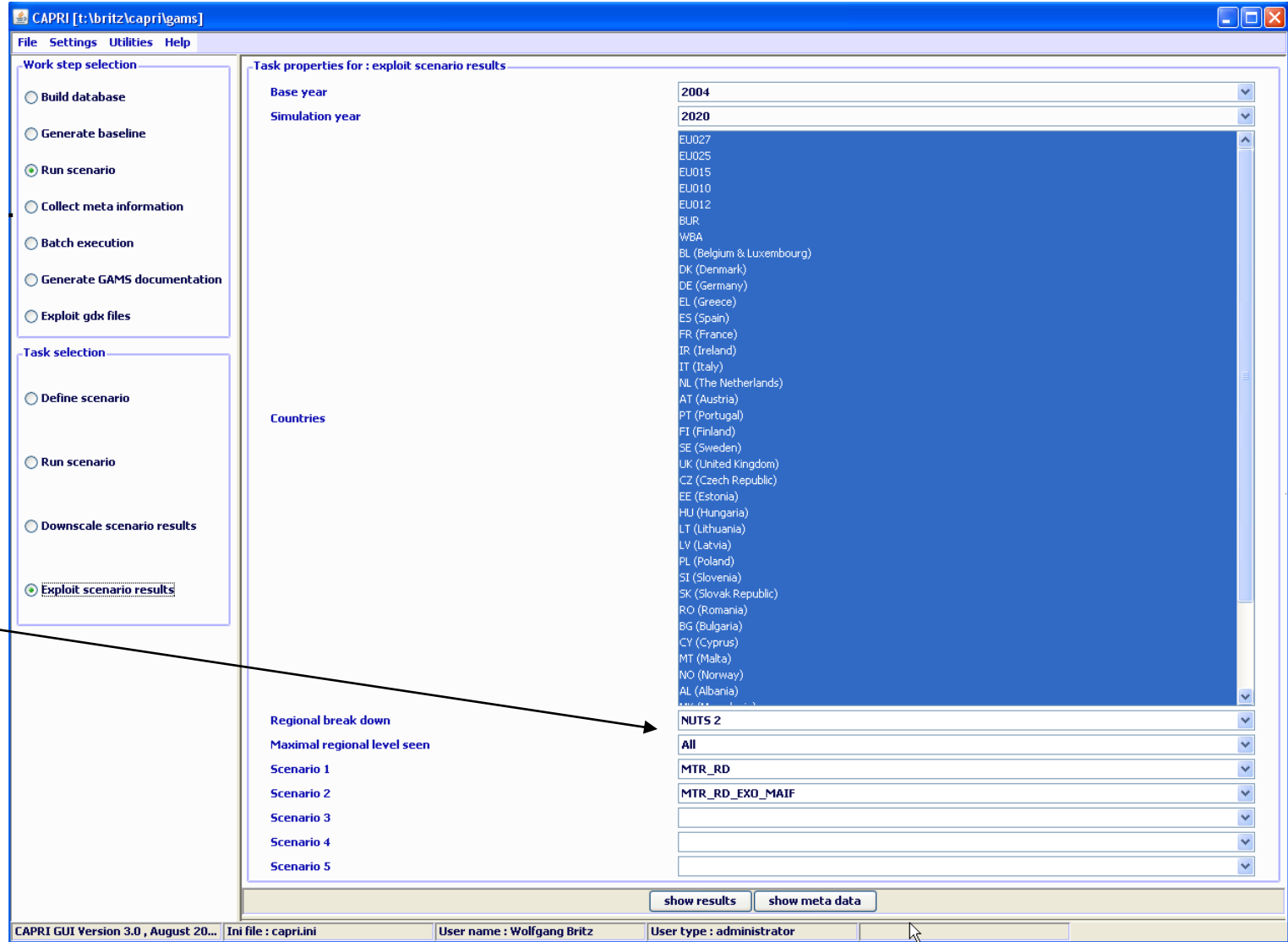
```

Bagsvaerdvej 246 A
DK-2880 Bagsvaerd, Denmark

Reading parameter(s) from "t:\britz\capri\gams\conopt.op4"
>> * -----
>> *   General Option file for CONOPT3
>> * -----
>> * Maximum and minimum jacobian element
>> rtmaxj = 1.00e+30
>> rtmaxv = 1.00e+30
>> * iteration log frequency
>> lfilos = 25
>> lfilog = 25
>> *rvhess = 10
>> lsimp = t
>> rtpiva = 1.E-15
>> rtnmai = 1.E-7
>> rtnma = 1.E-3
Finished reading from "t:\britz\capri\gams\conopt.op4"
Reading Data

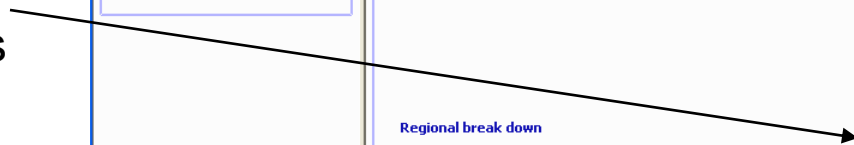
Iter Phase Ninf  Infeasibility  RQmax  NSB  Step InItr  MX  OK
0  0  9.9332362888E+02 (Input point)
      Pre-triangular equations: 743
      Post-triangular equations: 1120
1  0  9.9332362880E+02 (After pre-processing)
2  0  1.2218675248E+00 (After scaling)
    
```

The bottom status bar shows 'CAPRI GUI Version 3.0, August 2010', 'Ini file: capri.ini', 'User name: Wolfgang Britz', and 'User type:'.



After the run:
Exploit results ...

Chose
several
scenarios

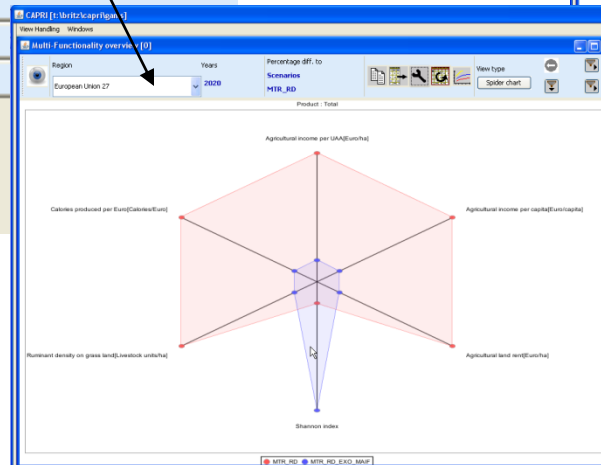
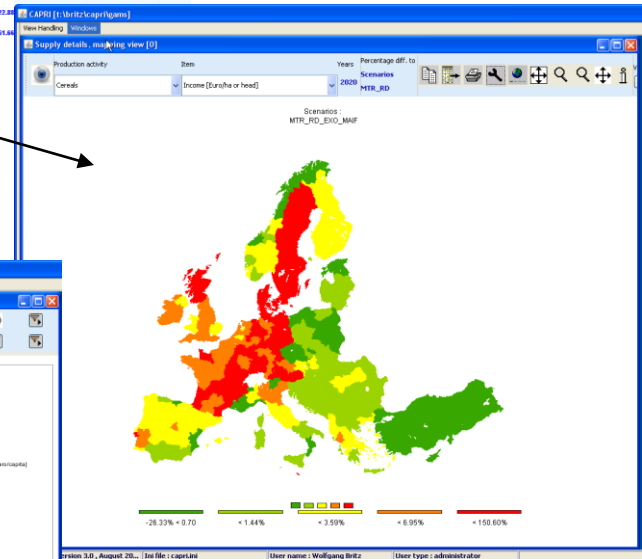


CAPRI GUI

Pre-defined **reports**
(tables, graphs, maps)

The screenshot shows the main menu of the CAPRI GUI. The 'CAP' category is highlighted, and a sub-menu is open showing options: 'Tax payers cost', 'Intervention stocks and their costs', 'Premium overview', and 'Premium details'. Other menu items include 'Meta', 'Welfare', 'Markets', 'Prices', 'BioFuels', 'Trade', 'Farm', 'Farm EU', 'HSMU', 'DNDC', 'Environment', 'Dual analysis', 'Multi-Functionality', and 'Energy'.

	Income [Euro/ha or head]	Hectares or herd size [1000 ha or heads]	Yield [kg or t/1000 hectares or head]	Supply [1000 t]
Cereals	346.74	58813.79	5753.68	
Grass	473.34	19125.80	2895.56	
Other arable crops	1126.21	8223.83	19922.85	
Vegetables and Permanent crops	6126.84	15893.23	11521.81	
Fodder activities	283.96	88251.38	22277.75	
Set aside and fallow land	157.58	15188.85		
All cattle activities	426.95	84291.69	94.16	
Beef meat activities	122.88			
Other animals	61.88			



Allows for pivot, selections, comparisons etc. ...

- Further GAMS related GUI features:
 - Generation of **HTML based documentation** of GAMS files à la Javadoc
 - **Batch processing** mode (check of all work steps after code changes, running packages of scenarios)
 - **Meta data generation/handling** (are stored along with numerical results in GDX files)

- GAMS has currently high market share for agricultural economic simulation models
- Probable reasons
 - Ease of use and convincing basic concept
 - Transparent link to solvers
 - Relatively low cost
 - Existing market penetration
- Larger projects wrap GAMS in applications

- Future market share of GAMS might depend on:
 - Further **performance gains** (parallel execution, „hot“ solver updates, ...)
 - **Combining** the appealing **simple concept** with **more advanced features** such as scoping, object oriented design, functions/subroutines ... but how?
 - Adding functionalities such as automated generation of project code documentation or link to SVN
 - **Development of alternatives** such as libraries in Object Oriented Languages (Java, C#) , already now applied for some newer methodologies such as Agent Based Modelling

Thanks for your attention

more on CAPRI

www.capri-model.org

contact author

Wolfgang.Britz@ilr.uni-bonn.de