PAUL SCHERRER INSTITUT

PSI

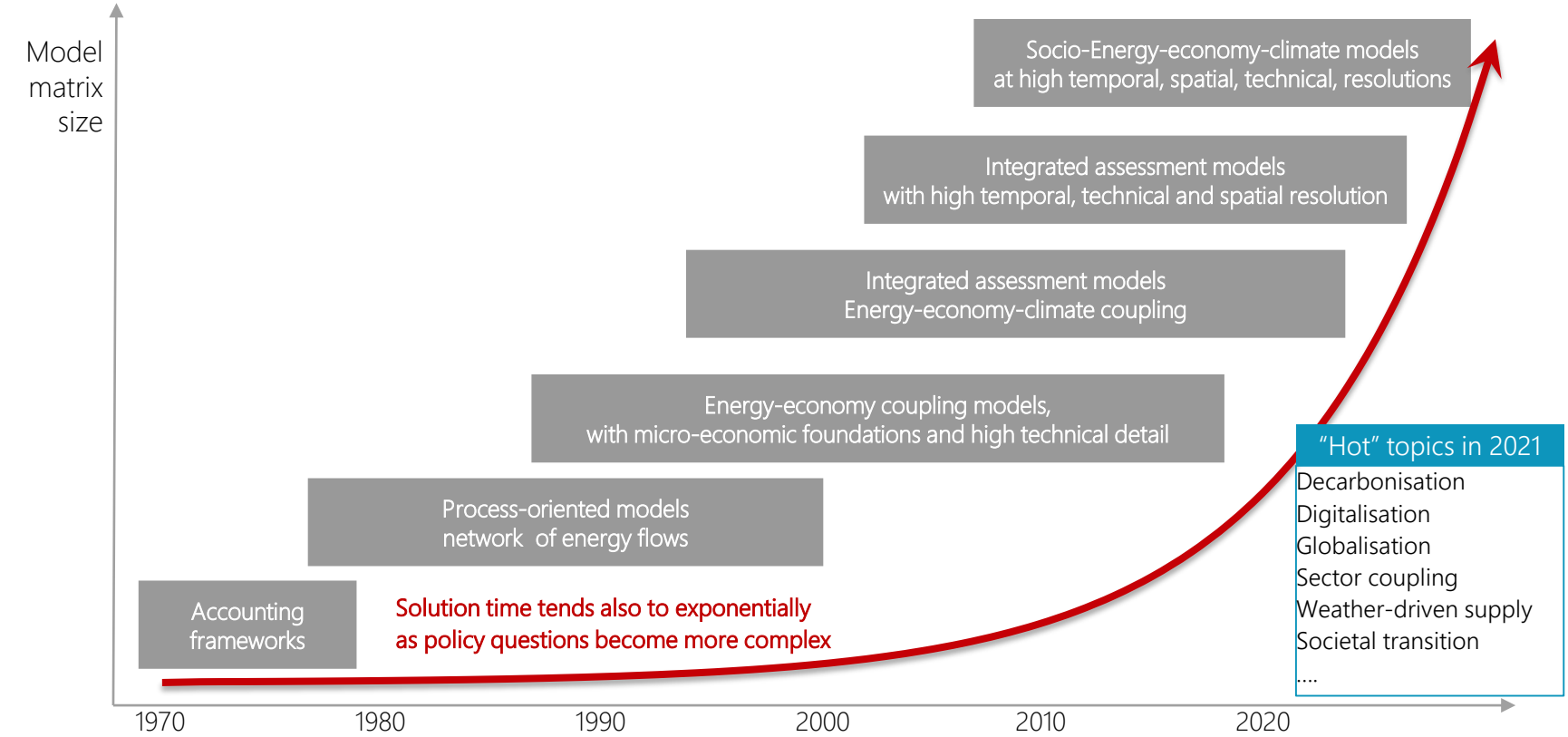
Evangelos Panos :: Paul Scherrer Institute

Aymane Hassan :: KTH Royal Institute of Technology

Acceleration strategies for energy systems models: Insights using the TIMES modelling framework

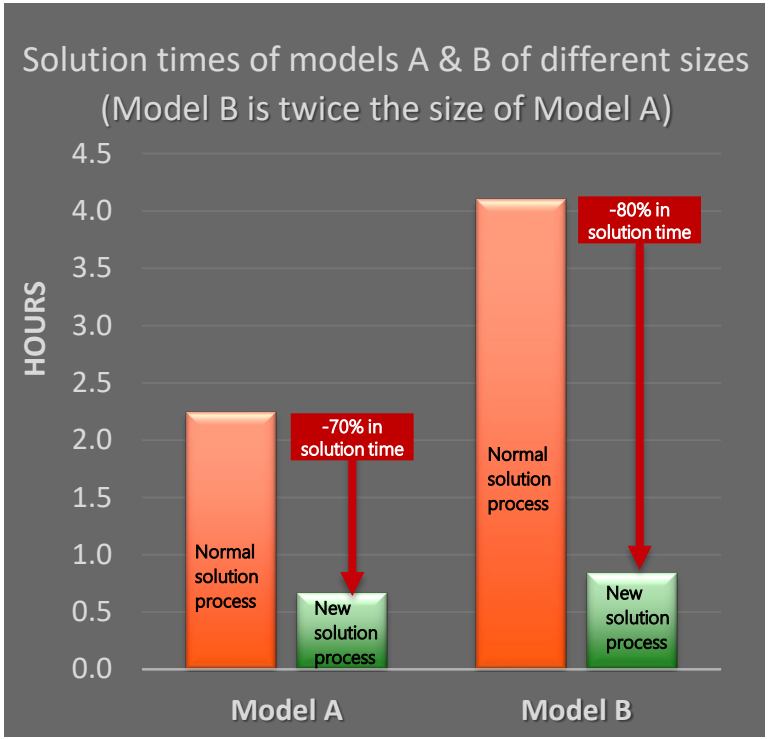
39th International Energy Workshop (IEW), Freiburg Germany, 14-17 June 2021

Need to deal with increasing complexity in energy systems analysis



Problem focus:	Energy balance frameworks	Energy markets and economy	Climate change mitigation	Consumers & sector coupling

Can we “break” the relationship between matrix size and solution time?



What do we know about our model matrix?

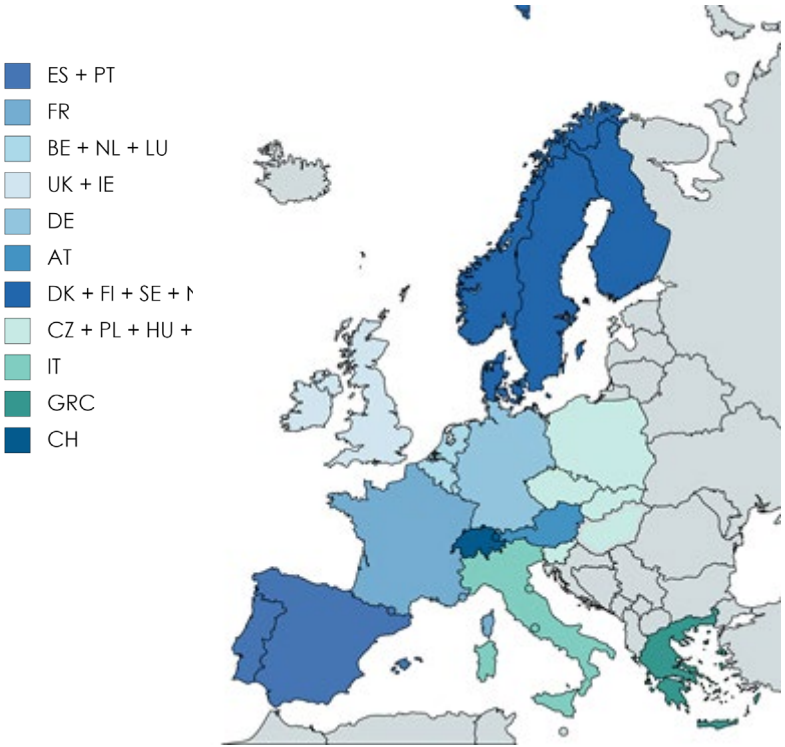
- It is (very) sparse
- Equations and variables are defined over dimensions:
  - Regions, technologies, timeslices, years, …
- Sub-matrices can be identified in model matrix:
  - E.g. all equations and variables of a particular “region”

Split the model matrix to submatrices and solve them in parallel at different CPUs

Project BEAM-ME

http://www.beam-me-projekt.de/beam-me/EN/Project/project_node.html

Using the TIMES energy modelling framework to test the concept

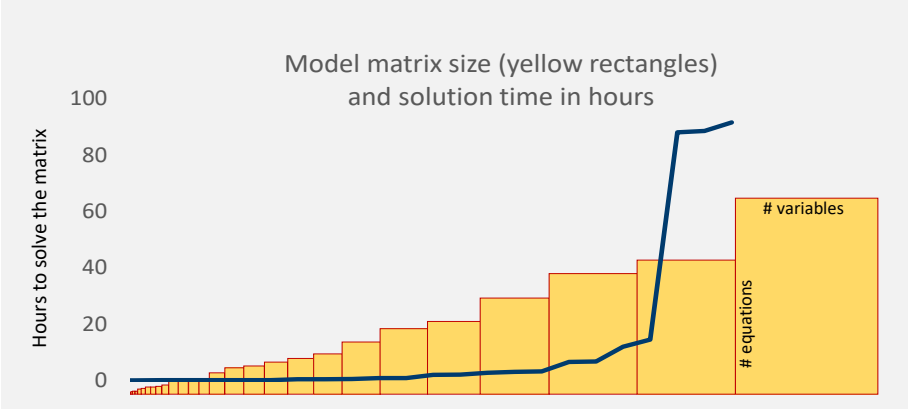


EUSTEM model of PSI: energy system model for Europe

- Technology rich bottom-up model
- Long term horizon from 2020 to 2080
- **Flexible** region definitions: from 1 to 22
  - default 11 → see map on the left
- **Flexible** intra-annual resolution: from 48 to 8760 timeslices
  - default 288: 4 seasons x 3 typical days x 24h
- Detailed unit commitment algorithm for power plants

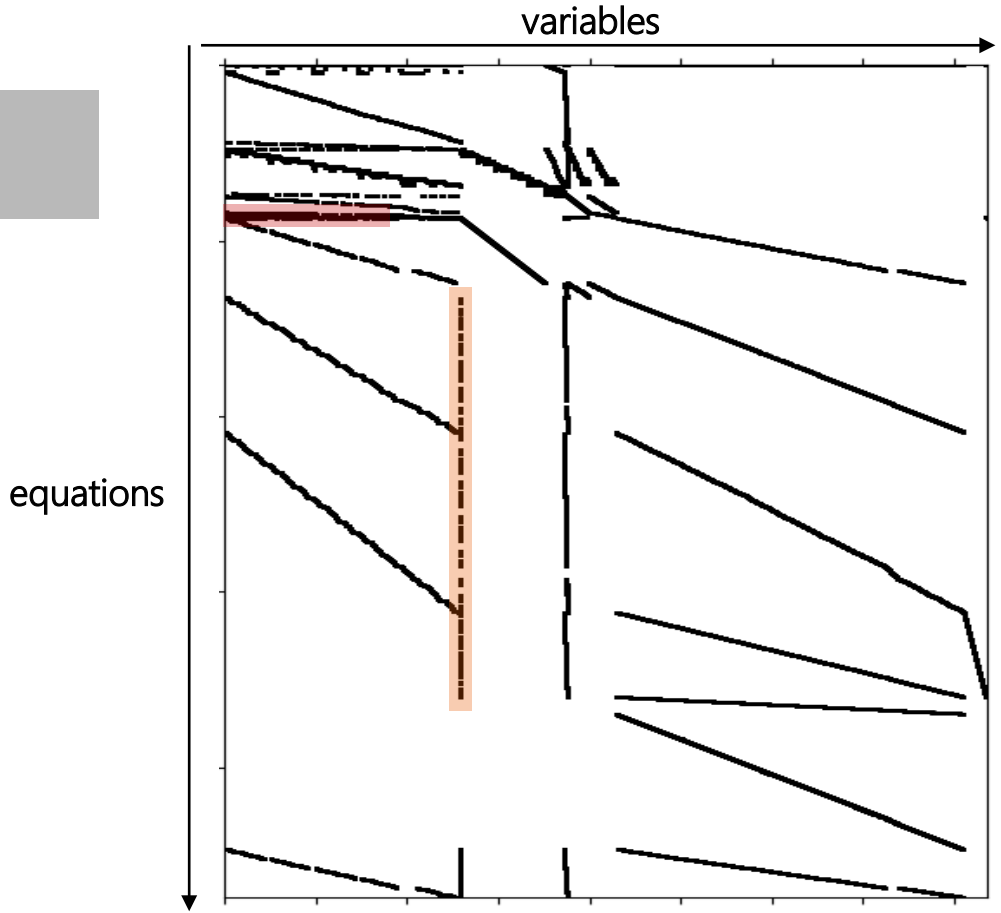
Creating instances of EUSTEM of different sizes and solution times

Instance name (timeslices_regions)	Equations (million)	Variables (million)	Non-Zeros (million)	Solution time (hours) (solver time only)*
48_1	0.2	0.2	1.1	0.003
48_2	0.4	0.3	2.1	0.007
….	…	…	…	…
288_11	12.3	8.4	75.5	2.2
288_22	22.8	15.6	138.8	4.1
….	…	…	…	…
672_22	52.7	36	320.5	88.1
…	…	…	…	…
2016_11	85.7	58.1	525.6	91.6



* Solution times obtained on a PC with:
GAMS version: 31.1.1
Solver: CPLEX/Barrier w/o crossover
CPU: 1 Intel Xeon Gold 6152
Paralller threads for CPLEX : 22

How the model matrix A of EUSTEM looks



EUSTEM algebraic representation:

$$\min \mathbf{c}^{\mathbf{T}} \mathbf{x}$$
$$\boldsymbol{A}\mathbf{x} \leq \mathbf{b}$$
$$\mathbf{x} \geq 0$$

- Each dot is an equation coefficient in A
- Very sparse matrix
- Coefficients occur in "blocks"
- "Linking" variables and constraints visible

Revealing the block structure of the model : annotation

1

```
set blocks number of blocks /bl1*bl35/;
```

Specify how many blocks (i.e. CPU nodes) will be used for solving the model

2

```
set blockassignment(blocks,allyear,all_reg,ALL_TS)
/
    bl1.(2020).(EST).(#all_ts)
    bl2.(2020).(NOR).(#all_ts)
    bl3.(2020).(STH).(#all_ts)
    bl4.(2020).(UKI).(#all_ts)
    bl5.(2020).(WST).(#all_ts)
                ...
    bl31.(2050).(EST).(#all_ts)
    bl32.(2050).(NOR).(#all_ts)
    bl33.(2050).(STH).(#all_ts)
    bl34.(2050).(UKI).(#all_ts)
    bl35.(2050).(WST).(#all_ts)
/;
```

Define the blocks across main dimensions of equations and variables of the model
(in this example, each block refers to a particular time period and region)

Block *bl35* contains all equations and variables defined for year *2050* and region *WST*

Revealing the block structure of the model : annotation

Using *.stage* to assign each variable to the one of the blocks; first block should be numbered as 2

3

```
var_act.stage(all_reg,v,all_year,p,all_ts) = sum(blockassignment(blocks, allyear,all_reg,all_ts), ord(blocks) + 1);

var_comnet.stage(all_reg,allyear,c,all_ts) = sum(blockassignment(blocks, allyear,all_reg,all_ts), ord(blocks) + 1);

var_comprd.stage(all_reg,allyear,c,all_ts) = sum(blockassignment(blocks, allyear,all_reg,all_ts), ord(blocks) + 1);
```

4

```
var_obj.stage(all_reg,obv,cur) = 1;

var_cap.stage(all_reg,allyear,p) = 1;
```

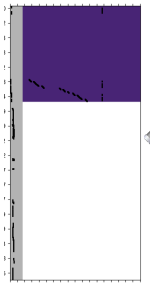
Variables existing across several blocks are “linking variables” and assigned to *.stage* = 1

- e.g. objective or capacity variables “living” many years, in a block decomposition based on “years” dimension

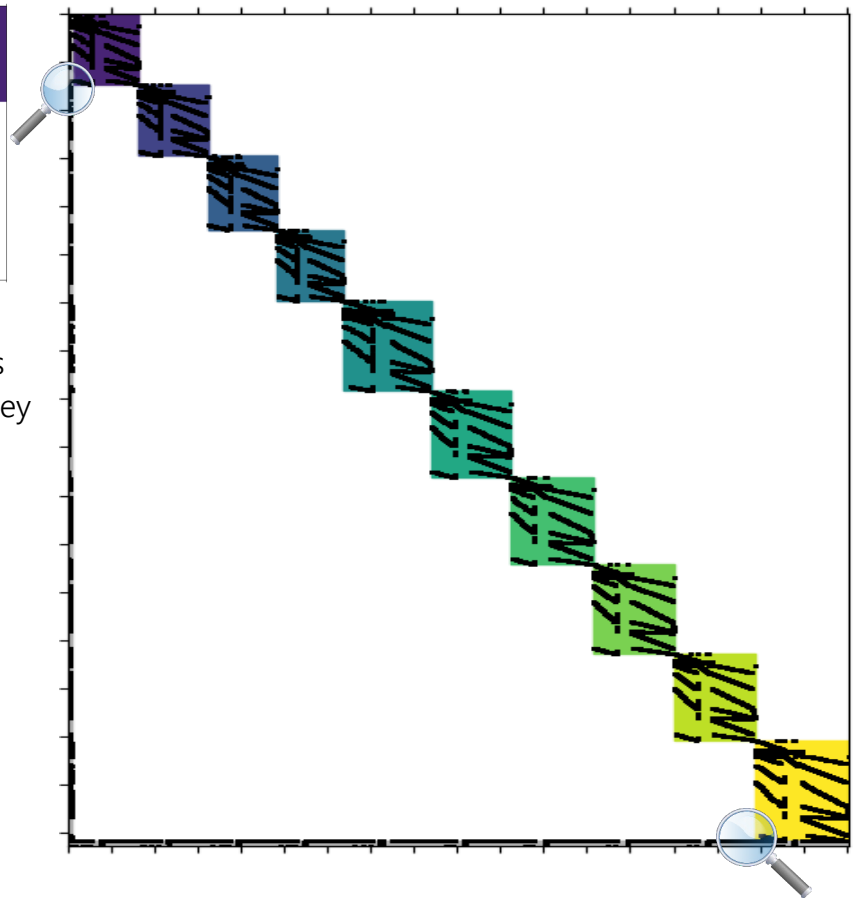
Same for equations, by setting *.stage* = number of block+1.

- equations having only “linking variables” are assigned to *.stage* = 1
- equations that contain variables from >1 blocks are assigned to *.stage* = cardinality(blocks)+1

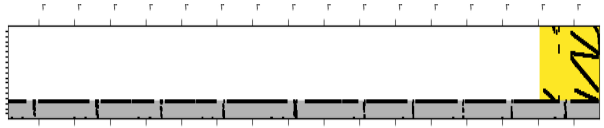
How the model matrix ***A*** of EUSTEM looks **after annotation**



Linking variables are in grey colour

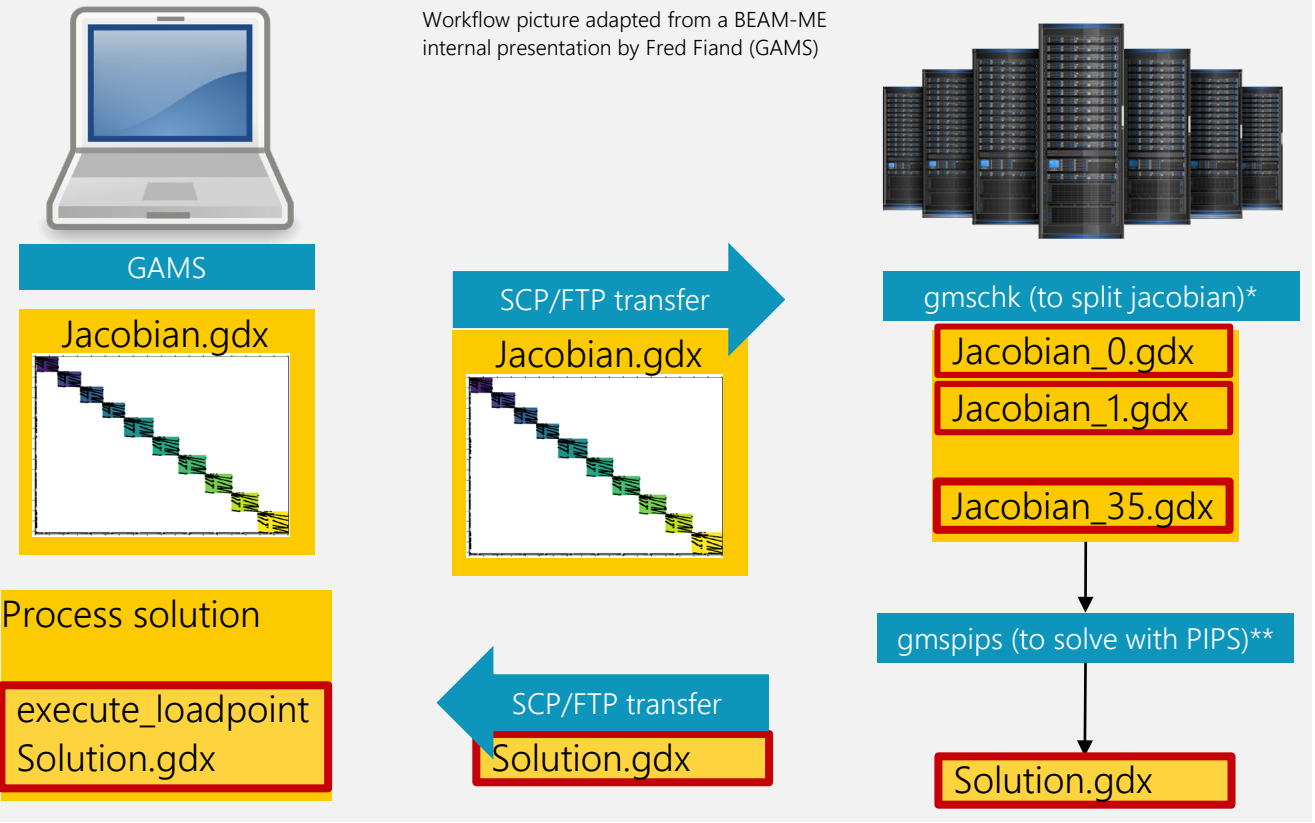


- Each dot is an equation coefficient in A
- Variables and equations are organised into distinct blocks, which are revealed to solver via the *.stage* attribute
- These blocks can now be solved in parallel at different CPUs



Linking equations are in grey colour

Split the annotated matrix and solve it with PIPS-IPM solver in HPC



PIPS is a suite of parallel optimisation solvers developed at Aargon

PIPS-IPM is a parallel interior point solver for LPs and convex QPs, further improved in the BEAM-ME project by Zuse Institute Berlin (ZIB)

The decomposition in PIPS-IPM is based on the Schur complement:
- but the computation of the Schur complement is not parallel

* gmschk –t –X –g «%gams.sysdir%» 36 jacobian.gdx
** srun –n «%number_tasks%» gmspips 36 jacobian.gdx «%gams.sysdir%

PIPS-IPM is under development and has limitations

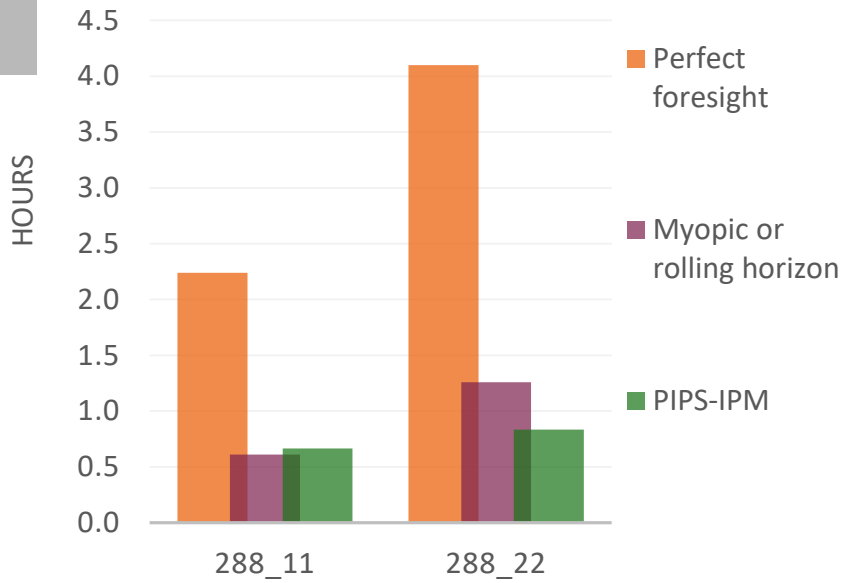
"Nothing worth having comes easy."

– Theodore Roosevelt

- Annotation of the model must comply with PIPS-IPM requirements:
  - Max number of total linking variables and equations
  - Max number of linking variables and equations per block
  - Avoid dense links of blocks and opt for 2-links (consecutive)
  - To avoid overhead in CPU communications blocks need to be many and of the same size
- PIPS-IPM is being improved in presolve and scaling methods
- PIPS-IPM is being equipped with hierarchical calculation of Schur Complement to handle models with strong linkages

Solving EUSTEM on PIPS-IPM - Conclusions

Solution times of two EUSTEM instances



Instance	Equations	Variables	Non-Zeros
288_11	12.3 mio	8.4 mio	75.5 mio
288_22	22.8 mio	15.6 mio	138.8 mio

- Not all instances of EUSTEM could be solved by PIPS-IPM
- Investigation of model to eliminate redundancies that could increase linking constraints
- Different annotations examined to comply with PIPS-IPM limitations
- Instance 288_11 was solved on 7 nodes on JSC supercomputer with 77 blocks in model matrix
- Instance 288_22 was solved on 10 nodes on JSC and also had 77 blocks in model matrix
- PIPS-IPM can solve large scale energy systems models **without changing** their mathematical problem (e.g. like myopic runs or typical model aggregation techniques over time and space do)

Wir schaffen Wissen – heute für morgen

My thanks go to:

- Daniel Rehfeldt (ZIB)
- Nils Kempke (ZIB)
- Manuel Wetzel (DLR)
- Frieder Borggrefe (DLR)
- Thomas Breuer (JSC)
- Fred Fiand (GAMS)

Contact:

Dr Evangelos Panos
Energy Economics Group
Paul Scherrer Institute
evangelos.panos@psi.ch

More info:
http://www.beam-me-projekt.de/beam-me/EN/Project/project_node.html