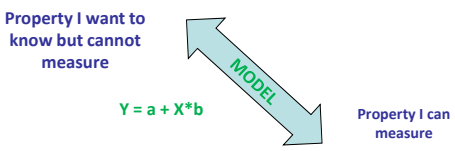
GEMSFITS:
A coupled code for GEMS input data optimization

George D. Miron, Dmitrii A. Kulik, Svitlana V. Dmytrieva, Thomas Wagner

Thermodynamic modeling

- Insight into chemical processes: chemical engineering, chemistry, materials research, and geochemistry
- Simple well controlled experimental systems → model more complex systems



Eidgenössische Technische Hochschule Zürich | Paul Scherrer Institut PSI | Fluids and Mineral Resources Group | GEMS Workshop EMPA 7-8 May 2014

Thermodynamic models

relations between thermodynamic properties,
calculation of measurable properties

+

Thermodynamic datasets

- actual values of thermodynamic properties

(compilation, derivation, consistency, errors, bookkeeping)

Simultaneous processing of data ensures that the derived properties are internally consistent and accurately reproduce the experimental data (Anderson and Crerar, 1993)

What do we need?

GEMSFITS

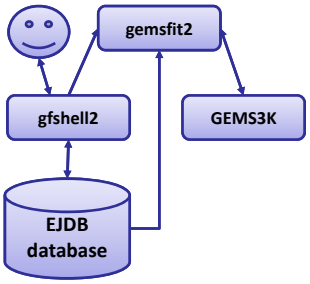
- Suite of appropriate models of mixing in phases → **TSolMod library**
- Efficient numerical method to perform thermodynamic computations → **GEMS3K**
- Optimization algorithms → **NLopt**
- Extensive, flexible collection of experimental data → **NoSQL (EJDB)**

ETH Eidgenössische Technische Hochschule Zürich | PAUL SCHERRER INSTITUT PSI | Fluids and Mineral Resources Group | GEMS Workshop EMPA 7-8 May 2014

GEMSFITS

GEMSFITS Code Package

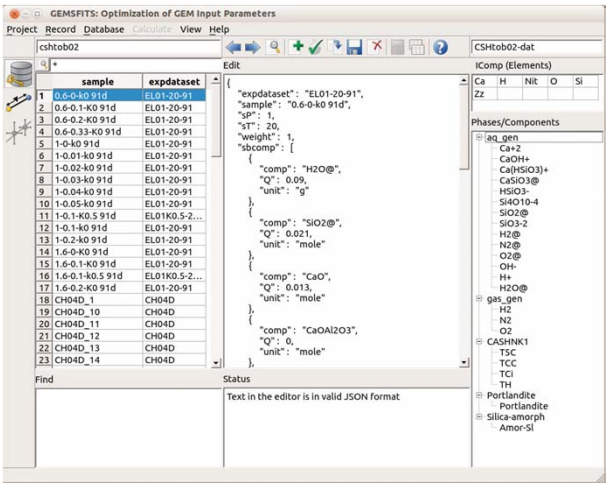
- Graphical user interface
- Experimental database management
- Setting up the fitting task
- Optimization algorithms
- Results and statistics



gfshell2: The GEMSFITS GUI

- **Database Mode**
- **Task Mode**
- **View Results Mode**

(more in the live example)



```
{
  "_id": "0x321c209",
  "sample": "Ben_1",
  "sT": 101.5,
  "sP": 0,
  "expdataset":"Bhm_Benezeth_etal2001",
  "sbcomp": [
     {
       "comp": "AlO(OH)",
       "Q": 10,
       "unit": "g" },
     {
       "comp": "H2O",
       "Q": 400,
       "unit": "g" },
     {
       "comp": "HCl",
       "Q": 0.0888514,
       "unit": "g" } ],
  "expphases": [
     {
       "phase": "aq_gen",
       "phprop": [
          {
            "prop": "pH",
            "Q": 2.484 } ],
       "phIC": [
          {
            "IC": "Al",
            "Q": 3.68129E-4,
            "unit": "molal" } ] } ]}
```


GEMSFITS database

<Key> : <Value>
"comp" : "AlO(OH)"

Spreadsheet → gfshell2 (csvtoejdb) → Database

- Flexible for various type of experiments
- Easy data selection for various optimization runs
- Select-Copy-Read-Update-Delete using the GUI in Database Mode
- **Stores: experiments, tasks, results**

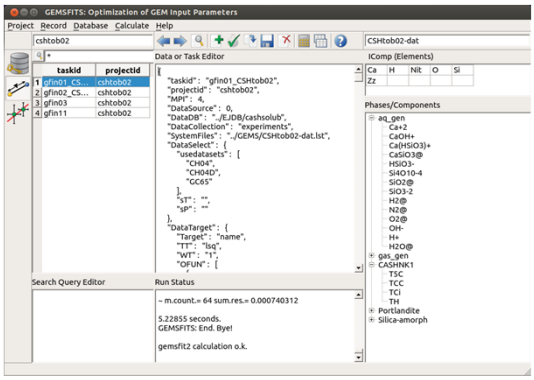
JSON

Setting up the fitting task

(in the task specification file)

- **What data to use:** select/exclude experiments (samples) from database
- **What to fit:** marking the adjustable parameters
- **What to compare:** solubility, solid-solution composition, pH, Eh, activities, etc.
- **How to calculate the residuals:** e.g. least squares, weighting

GEMSFITS GUI in Task Mode with task specification editor



JSON

What can you fit?

- Non-ideal solution **activity model parameters**
- **G˚ values** of dependent components (e.g. aqueous species, solid solution end members, pure condensed substances)
- **Temperature** and/or **pressure, bulk chemical composition** for inverse modeling tasks

Constraint options

- Upper and lower bounds on freely-adjustable parameters
- Parameters can be declared as freely adjustable ‘F’; linearly constrained ‘L’; reaction-constrained ‘R’

Fitting parameter markup in JSON

```
<PMc>, <G0>, <bIC>,
<TK>, <P>
...
F{
     "IV" :   -10300,
     "LB" :   -16000,
     "UB" :   -6000
} ...

...
F-10300
...
```

```
<G0>

...
R{
     "IV" :   -696322,
     "Ref" :   "We93",
     "order" :   "1",
     "nC" :   4,
     "rDC" :   "AlOH+2",
     "RC" :   [
          "Al+3",
          "H2O@",
          "H+",
          "AlOH+2"
     ],
     "Rcoef" :   [
          -1,
          -1,
          1,
          1
     ]
}
...
```

```
<bIC>

...
L{
     "LE" :   "H",
     "IV" :
44.5761406330092,
     "LEs" :   [
          "Na",
          "Cl"
     ],
     "Lcoef" :   [
          1,
          1
     ]
}
...
```

ETH Eidgenössische Technische Hochschule Zürich | PAUL SCHERRER INSTITUT PSI | Fluids and Mineral Resources Group | GEMS Workshop EMPA 7-8 May 2014

Objective function

- Determines what experimental data is compared with its computed counterpart
- Many objective function terms → different type of data can be used simultaneously

Nested target function

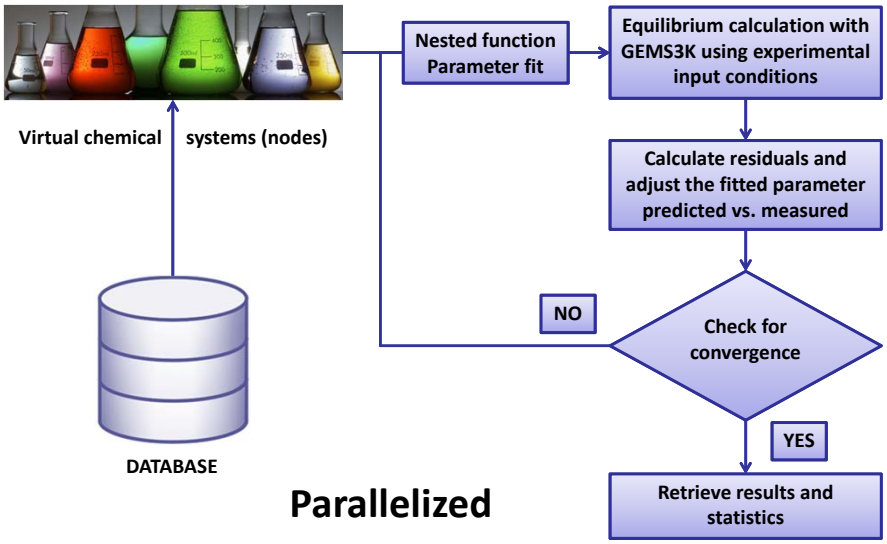
- Adjust parameters specific to each experiment after each optimization step (e.g. adjust pH)
- Used for inverse modeling (e.g. adjusts T and P to a given phase composition for each sample)

Objective and nested functions in JSON

```
{    "Target" :   "name",
     "TT" :   "lsq",
     "WT" :   "1",
     "OFUN" :   [
          {
               "EPH" :   "aq_gen",
               "CT" :   "IC",
               "CN" :   "Al",
               "unit" :   "log_molal"
          }
     ],
     "NFUN" :   [
          {
               "Ptype" :   "bIC",
               "EPH" :   "aq_gen",
               "CT" :   "prop",
               "CN" :   "pH",
               "unit" :   "-loga"
          }
     ]
}
```

```
{    "Target" :   "name",
     "TT" :   "lsq",
     "WT" :   "1",
     "OFUN" :   [
          {
               "EPH" :   "aq_gen",
               "CT" :   "IC",
               "CN" :   "Si",
               "unit" :   "log_molal"
          },
          {
               "EPH" :   "aq_gen",
               "CT" :   "IC",
               "CN" :   "Ca",
               "unit" :   "log_molal"
          }
     ],
     "ADDOUT" :   [
          {
               "SRC" : "meas",
               "EPH" : "CSH",
               "CT" : "prop",
               "CN" : "Q",
               "unit" : "mol"
          }
     ]
}
```

gemsfit2 code flowchart



Weighting & outliers

- Set in the database of experiments or in the task configuration
- Set by the user for each sample, for each objective function term (type of data)
- Using the measurement error, measured value, measurement average
- Give outliers zero or low weight
- Tukey's Biweight: uses a threshold value (*C*) and gives samples which residuals exceed it a zero weight, and the rest - a weight as function of the distance to *C*.

Optimization algorithms

- Various global and local algorithms implemented in Nlopt library (http://ab-initio.mit.edu/nlopt/)
- Global algorithms – large search space, slower, converges to the "true" minimum value
- Local algorithms – small search space, much faster, can converge to local minimum values
- Start global, continue with local for higher precision

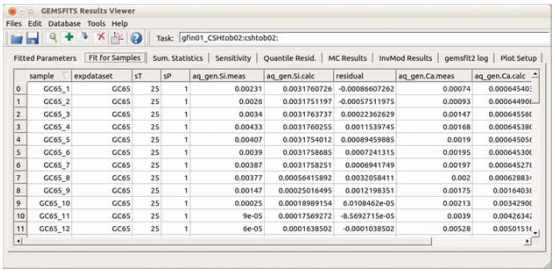
Statistics

- Goodness of fit, sensitivity analysis, correlation coefficients
- Parameter confidence intervals using

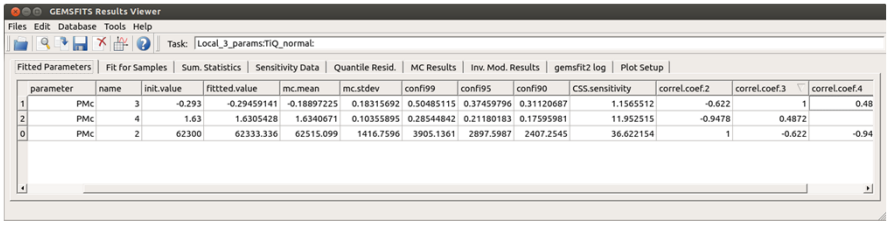
  Monte Carlo simulations
  - Uses the experimental data scatter to create many random simulated measured data
  - Finds the best fit value of parameters for each set
  - Confidence intervals computed form the standard deviation of parameters obtained from each data set

Output

- Output files in csv format read in by the GUI to View Results window tabs
- Can be saved in the database



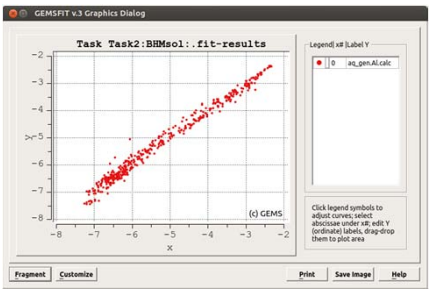
GEMSFITS GUI View Results Mode



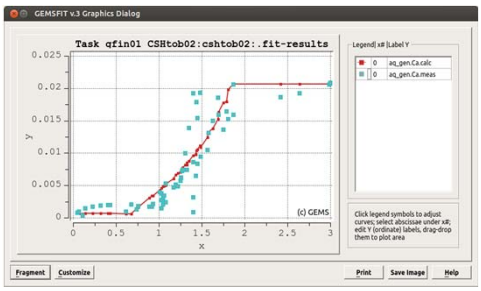
Output

- Values from the output tabs can be plotted in XY coordinates

measured Vs. calculated (Al solubility)



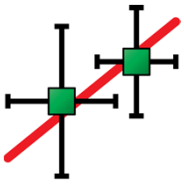
CSH amount Vs. measured & calculated Ca solubility



Outlook

- Plans to make GEMSFITS available for Windows (currently Linux only)
- Extensions to fit parameters of sorption models and mineral dissolution / precipitation kinetics models (TSorpMod and TKinMet libraries)
- Fit GEM-Selektor built-in equation-of-state parameters
- MongoDB server database (large data collection of various experiments) accessible to many scientists → optimizing TDBs and models for a large number of applications

Thank you!
And now let us try GEMSFITS!



- This work is funded by ETH-19 11-2 grant at ETHZ IGP and is part of the GEMS TM project at LES PSI
- Thanks to Professor Ch.Heinrich (IGP) for support and discussions