[IrisToolbox] for Macroeconomic Modeling

Nonlinear Equations Solver

`jaromir.benes@iris-toolbox.com`

Use of the solver

- Calculate steady state
- Run nonlinear stacked-time simulations
- Run nonlinear period-by-period simulations

IrisT implementation

- Augmented Quasi-Newton-steepest-descent (QNSD) algorithms
- Analytical (symbolic) or numerical Jacobian
- Sequential block pre-analysis
- Detection of sparse Jacobian pattern
- Detection of invariant Jacobian elements

Steepest descent (Cauchy) step

- Flavor of global optimization
- Robustness
- Underdetermined systems, aka singularity in Jacobian: critical for growth models

Curse of dimensionality in stacked-time simulations

Dimension of the problem: $K =$ number of equations $\times$ number of periods

Dimensions of the Jacobian: $K \times K$

The conventional ways of handling terminal condition require a larger number of periods to be simulated (to discount the effect of the "wrong" terminal condition)

Reduce the actual dimensionality and accelerate:

- Base terminal condition upon the first order solution dramatically reduces the number of periods needed
- Detect the sparse Jacobian pattern to avoid zero points
- Detect the Jacobian elements that need to be evaluated only once (at the beginning)

QNSD algorithm

Quasi-Newton-steepest-descent where the Jacobian is regularized using the steepest descent method for underdetermined systems (steady state solver for growth models with "independent" degrees of freedeom, or steady state solver with the `"fixLevel"` option)

$$x_k = x_{k-1} - s_k \, D_k$$

$$D_k = \left(J_{k-1}^T \, J_{k-1} + \lambda_k \right)^{-1} J_{k-1} \, F_{k-1}$$

where

- function evaluation $F_k = F(x_k)$
- Jacobian evaluation $J_k = J(x_k)$
- step direction D_k
- step length s_k
- steepest descent (Cauchy step) mixin parameter λ_k

Special cases

- Quasi-Newton with variables step length: $\lambda_k = 0$
- Plain vanilla Newton $\lambda_k = 0$, $s_k = 1$

The `simulate` function

```
[s, info, frameDb] = simulate( ...
    m, d, range, ...
    "deviation",    true | false, ...
    "anticipate",   true | false, ...
    "plan",         empty | Plan, ...
    "method",       "stacked", ...
    "blocks",       true | false, ...
    "terminal",     "firstOrder" | "data", ...
    "startIter",    "firstOrder" | "data", ...
    "successOnly",  false | true, ...
    "window",       @auto | numeric, ...
    "solver",       @auto | {"iris-newton", ...}, ...
);
```

The `solver` options:

```
{ "iris-newton", ...
    "skipJacobUpdate",          0 | numeric, ...
    "lastJacobUpdate",          Inf | numeric, ...
    "functionNorm",             2 | Inf, ...
    "maxIterations",            5000 | numeric, ...
    "maxFunctionEvaluations",   @auto | numeric, ...
    "functionTolerance",        1e-12 | numeric, ...
    "stepTolerance",            1e-12 | numeric, ...
}
```